

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**

(19) BUNDESREPUBLIK

DEUTSCHLAND



DEUTSCHES

PATENTAMT

# Offenlegungsschrift

(10) DE 44 22 837 A 1

(51) Int. Cl. 5:

G 06 T 5/00

B 41 J 2/485

H 04 N 1/409

40049/IV

(30) Unionspriorität: (32) (33) (31)

30.06.93 JP 5-187058

(71) Anmelder:

Ricoh Co., Ltd., Tokio/Tokyo, JP

(74) Vertreter:

Schwabe, H., Dipl.-Ing.; Sandmair, K., Dipl.-Chem.  
Dr.jur. Dr.rer.nat.; Marx, L., Dipl.-Phys. Dr.rer.nat.,  
Pat.-Anwälte, 81677 München

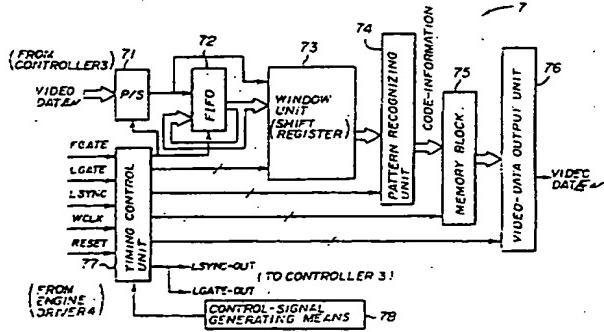
(72) Erfinder:

Ohshita, Masakazu, Kawasaki, Kanagawa, JP;  
Shimomae, Mutsuo, Yokosuka, Kanagawa, JP

Prüfungsantrag gem. § 44 PatG ist gestellt

(54) Bilddaten-Verarbeitungssystem

(57) Eine Bitmapeinheit (2) erzeugt ein Bitmap mit Hilfe vorgegebener Bilddaten und eine Grenzen erkennende Einheit (74) erkennt Kenndaten einer Grenze zwischen einer ersten und einer zweiten Pixelart, welche das Bitmap bilden. Eine Bestimmungseinheit (74) bestimmt für jedes Pixel, ob eine Korrektur notwendig ist, um das aus den vorgegebenen Bilddaten erzeugte Bild merklich feiner zu machen. Eine Korrekturseinheit (75) führt eine Korrektur an den Bilddaten für das Pixel durch, dessen Korrektur als notwendig festgestellt worden ist. Eine Datenwiederholeinheit (72) wiederholt die vorgegebenen Bilddaten, die bei der von der Korrekturseinheit durchzuführenden Korrektur zu verwenden sind. Eine Einheit (78) erzeugt eine Zeitsteuerung, innerhalb welcher die Datenwiederholeinheit die Wiederholoperation durchführt. Eine Einheit (78) setzt die Anzahl Mal fest, bei welcher die Datenwiederholeinheit die Wiederholoperation durchführt. Eine Numerierungseinheit (78) numeriert den Bilddatensatz, welcher bei der Wiederholung für jede Wiederholoperation erzeugt worden ist, um die Zahl jedes bei der Wiederholung erzeugten Bilddatensatzes zu identifizieren.



DE 44 22 837 A 1

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

BUNDESDRUCKEREI 12.94 408 067/459

58/29

DE 44 22 837 A 1

## Beschreibung

Die Erfindung betrifft ein Bilddaten-Verarbeitungssystem und -verfahren, welche in einer Bildverarbeitungseinrichtung mit einem elektrophotographischen System, das digitale Bilddaten verarbeitet, wie ein optischer Printer/Drucker, wie ein Laser-Printer, ein Digitalkopierer, ein Seitendruck-Faksimilegerät oder eine Bildanzeigevorrichtung, verwendet werden können. Die Erfindung betrifft insbesondere eine Bildverfeinerung in solchen Geräten/Systemen.

Derartige Bilderzeugungs- und Bildanzeigeeinrichtungen quantisieren Textbilddaten oder Abbildungsbilddaten, so daß die relevanten Bilddaten in das Punkt-matrix-Bitmap-Format mit Zweitondaten gedehnt werden. Die Textbilddaten werden durch Umsetzen der Textcodedaten einschließlich Schriftaritäten erhalten, und die Abbildungsbilddaten werden durch Lesen mit Hilfe eines Bildscanners u. a. erhalten. Das Bitmap-Format der relevanten Daten wird in einem Videospeicher-Bereich in einem RAM (Random Speicher) gespeichert und wird dann nacheinander ausgelesen, um als Videodaten einer Bilderzeugungseinheit oder einer Bildanzeigeeinheit zugeführt zu werden. Die Bilderzeugungseinheit erzeugt das entsprechende Bild auf einem Aufzeichnungsmedium, wie einem Blatt Papier u. a., und die Bildanzeigeeinrichtung stellt das entsprechende Bild auf einem Bildschirm dar.

Ein ideales Bild ist ein analoges Bild, dessen Umriß sich in einer beliebigen Richtung kontinuierlich ausdehnen kann. Ein derartiges Bitmap-Bild, das als ein Quantisierungsergebnis erhalten worden ist, ist ein digitales Bild, dessen Umrißlinie sich in den vorherbestimmten zwei zueinander senkrechten Dimensionen der Punkt-matrix ausdehnen kann. Wenn ein Umriß des digitalen Bildes, welcher schräg oder bogenförmig verläuft, ausgedrückt werden soll, muß die Umrißlinie abgestuft entlang der durch die Punktmatrix vorgegebenen Linien (wobei eine solche Erscheinung auch als "Zacke" ("jag") bezeichnet werden kann) im allgemeinen schräg oder bogenförmig verlaufen. Diese unvermeidliche Eigenschaft des digitalen Bildes kann die Feinheit des endgültigen Bildes mindern, d. h. kann eine genaue Wiedergabe eines Vorlagenbildes oder eine genaue Darstellung einer gewünschten Umrißlinie vereiteln.

Eine Verkleinerung in der Punkt-(Pixel)-Größe der Punktmatrix, d. h. ein Erhöhen der Anzahl Punkte, die in einer Flächeneinheit vorhanden sind, (ein Erhöhen der Punktmatrix-Auflösung) kann eine derartige Bildverschlechterung mindern. Jedoch werden durch eine solche Auflösungszunahme die Kosten beträchtlich erhöht. Um beispielsweise ein  $600 \times 600$  dpi-zweidimensionales Bitmap als Ergebnis des Verdoppelns der Auflösung von  $300 \times 300$  dpi zu erhalten, muß die Speicherkapazität um das 4-fache und die Verarbeitungsgeschwindigkeit bei der Datenverarbeitung ebenfalls um 4-fache erhöht werden.

Andere Maßnahmen, eine Bildverschlechterung zu mindern, betreffen eine Interpolations-Technologie, bei welcher eine Verbindungslinie zwischen benachbarten Winkelkanten, die eine abgestufte Umrißlinie darstellen, erzeugt wird, um so eine Schräge dazwischen zu bilden. Gemäß einem anderen Aspekt der Interpolations-Technologie wird die Helligkeit zwischen benachbarten Umrißpunkten "geglättet" bzw. ausgeglichen, um so die Kante undeutlich zu machen. Durch solche Maßnahmen werden Zacken an der Umrißlinie wirksam geglättet, jedoch können Kontraste und/oder die Auflösung

schlechter werden, da feine Formen undeutlich gemacht werden.

Um diese Schwierigkeit zu lösen, ist in dem US-Patent 4 544 922 eine Technologie entwickelt worden. Gemäß dieser Technologie wird ein Punkt mit einer Größe, welche kleiner als die ursprüngliche Größe ist, hinzugefügt, oder ein Teilbereich mit der vorerwähnten kleineren Punktseite wird aus einem entsprechenden Teil eines Punktmusters entfernt, das durch ein Bitmap dargestellt ist, um den relevanten Teil zu korrigieren. Eine Mustererkennungs-Technologie und/oder eine Formenvergleichs-Technologie (template comparison technology) werden verwendet, um einen in dem Punktmuster zu korrigierenden Teil festzustellen. Bei der vorstehend angeführten Technologie wird der Mustererkennungs- oder der Formenvergleichs-Prozeß an allen Positionen eines Bitmap-Bildes und eine Punktkorrektur, wie vorstehend beschrieben, in angemessener Weise durchgeführt. Obwohl im Ergebnis eine Bildqualität (Bildfeinheit) durch Glätten einer Linienform verbessert werden kann, ohne Kontraste zu verschlechtern, ist das relevante Verarbeitungssystem extrem kostspielig und erfordert sehr lange (Verarbeitungs-)Zeit.

Um die Schwierigkeit zu lösen, ist in der offengelegten japanischen Patentanmeldung Nr. 2-112 966 ein Verfahren vorgeschlagen, bei welchem ein relevantes Bitmap-Bild mit einer vorher gespeicherten, vorherbestimmten Form (template) für jedes kleine Bruchteil der Bilder verglichen wird. Wenn dann eine Kenndaten-Anpassung zwischen dem Bitmap-Bild und einer Form für einen Bildbruchteil festgestellt wird, wird der Bruchteil des Bitmap-Bildes durch Korrekturpunkte korrigiert. Folglich ist die Bildqualität in gedruckten Bildern verbessert.

Zur Realisierung der vorstehend beschriebenen Methode wird beispielsweise ein nachstehend beschriebener Formen-Anpassungsprozeß durchgeführt. Die Daten des Bitmap werden in serielle Daten umgewandelt, um so in einen FIFO-Puffer eingegeben zu werden, und es wird eine Untermenge des Bitmap-Bildes aus  $N \times M$ -Bits gebildet. Ein Abfragefenster wird gebildet, um dadurch Daten aus der Untermenge von Bitmap-Bildern zu beobachten oder zu extrahieren, wobei das Fenster eine vorherbestimmte Form hat, eine vorherbestimmte Anzahl von Bits hat und ein zentrales Bit hat, welches demjenigen des Untermengen-Bildes entspricht. Dann werden die beobachteten/extrahierten Daten verwendet, um eine Anpassung mit Hilfe von Formen, welche verschiedene charakteristische Muster haben, festzustellen, welche Muster solche von zu korrigierenden Bildern sind.

Wenn eine Form zu den beobachteten/extrahierten Daten paßt, wird eine bestimmte Korrektur-Unterzelle (ein Korrekturpunkt) verwendet, um das zentrale Bit in dem relevanten Untermengen-Bild zu ersetzen. Die Korrektur-Unterzelle entspricht der relevanten Form. Wenn keine Form zu den beobachteten/extrahierten Daten paßt, wird das relevante zentrale Bit unverändert belassen.

Solche Formenanpassungsprozesse werden durchgeführt, indem nacheinander das zu verarbeitende Untermengen-Bild verschoben wird, so daß der gesamte Bereich des relevanten Bitmap-Bildes verarbeitet wird. Die Prozesse werden so durchgeführt, daß jedes Bit des Bitmap-Bildes nacheinander dem vorerwähnten zentralen Bit zugeordnet wird. Folglich kann im Vergleich zu der vorstehend angeführten, in dem US-Patent beschriebenen Technologie ein feines endgültiges Bild er-

halten werden, indem die Bildqualität sogar mit einer verhältnismäßig kleinen Speicherkapazität und mit einem relativ geringen Datenverarbeitungsaufwand verbessert wird.

Jedoch auch bei der vorstehend beschriebenen Methode sind eine große Anzahl Formen (tempates) bei der vorerwähnten Abfragefenster-Formation entsprechend allen charakteristischen Bildmustern notwendig. Anpassungsmuster, die in einem relevanten vorgegebenen Bitmap-Bild enthalten sind, sollten korrigiert werden. Wenn solche Muster entsprechend allen beliebigen Bildformen vorgesehen werden sollten, wird die Musteranzahl beträchtlich sein. Das Produzieren einer Anzahl Formen erfordert eine beachtliche Arbeitskraft und folglich beträchtliche Kosten. Ferner ist eine beträchtliche Speicherkapazität erforderlich, um die Anzahl Formen zu speichern. Ferner erfordern die vorstehend beschriebenen Formen-Anpassungsprozesse beträchtliche Zeit.

Um die Schwierigkeiten zu lösen, hat einer der Erfinder der vorliegenden Erfindung zusammen mit anderen Personen ein Bilddaten-Verarbeitungsverfahren und eine Einrichtung zu dessen Durchführung in der offengelegten japanischen Patentanmeldung Nr. 5-207 282 beschrieben.

Mit dem vorerwähnten Verfahren werden Zacken, wie sie eingangs beschrieben worden sind, korrigiert, um eine Bildqualität zu verbessern, indem eingegebene Bilddaten in einer Bitmap-Formation verwendet werden. Das Verfahren kann Daten reduzieren, welche vorher in einem Speicher gespeichert werden sollen, wobei die Daten verwendet werden, um einen entsprechenden Korrekturprozeß an dem Bitmap-Bild durchzuführen. Bei dem Verfahren werden Punkte in den eingegebenen Bilddaten, die zu korrigieren sind, und entsprechende Korrekturwege festgelegt, welche bei den eine Korrektur erforderlichen Punkten angewendet werden. Bei dem Verfahren wird die vorerwähnte Festlegung mittels einfacher logischer Operationen in einem sehr kurzen Zeitabschnitt mit Hilfe eines Mikroprozessors durchgeführt.

Dieses Verfahren wird nunmehr kurz beschrieben. Bei dem Verfahren werden die Kenndaten der Linienform der Grenze zwischen einem Schwarzpunkt- und einem Weißpunkt-Bereich in dem Bitmap von eingegebenen Bilddaten erkannt. Im Ergebnis werden dann die Kenndaten in einen Multi-Bitcode für jeden Punkt der eingegebenen Bilddaten umgesetzt. Dann wird zumindest ein Teil des Codes verwendet, um zu bestimmen, ob der relevante Punkt ein zu korrigierender ist oder nicht. Wenn dann bestimmt wird, daß der Punkt zu korrigieren ist, wird er entsprechend der Umstände korrigiert, welche in Abhängigkeit von dem relevanten Code bestimmt worden sind. Bei der vorstehend beschriebenen Festlegung wird ein Musterspeicher mit Adressen verwendet, welche den Codes entsprechen, wobei die entsprechenden Umstände, die für die Korrektur anzuwenden sind, entsprechend den Adressen festgelegt werden können.

Jeder der vorerwähnten Codes enthält einen Code, welcher die Schrägenrichtung der Linie an der Position des relevanten Punktes anzeigt, einen Code, welcher die Art der Schräge anzeigt, und einen Code, welche eine Position des relevanten Punktes anzeigt. Die vorerwähnte Position ist eine, deren Ursprung der Anfangspunkt in der Reihe von Punkten ist, welche ordentlich oder horizontal angeordnet sind und den relevanten Punkt enthalten.

Bei dem vorstehend beschriebenen Verfahren ist es

nich: notwendig, als Modelle alle charakteristischen Muster vorzusehen und zu speichern, welche den zu korrigierenden entsprechen. Mit dem Verfahren können zu korrigierende Punkte sowie die Art und Weise der Korrektur für eine Korrektur erforderliche Punkte in einfacher Weise und in einer kurzen Zeitspanne mit Hilfe der Codes bestimmt werden, welche, wie oben beschrieben, leicht erzeugt werden können.

Ferner werden Punktdaten von Punkten, die in einem Bereich vorhanden sind, dessen Mitte der relevante Punkt ist, über ein entsprechendes Fenster extrahiert. Der Bereich wird in einen Kernbereich, der in der Mitte angeordnet ist, und eine Anzahl peripherer Bereiche aufgeteilt, die peripher um den Kernbereich herum angeordnet sind. Der vorerwähnte Code wird basierend auf eine Erkennungsinformation erzeugt, die als ein Ergebnis erhalten worden ist, die Linienform zu erkennen, wie vorstehend erwähnt ist. Die Erkennungsinformation, die zum Erzeugen des Codes zu verwenden ist, ist die Kombination von Kernbereich- und Peripheriebereich-Erkennungsinformation. Die Kernbereichs-Erkennungsinformation wird mit Hilfe des Kernbereichs der Bilddaten und die Peripheriebereichs-Erkennungsinformation wird zumindest mit Hilfe eines peripheren Bereichs der Bilddaten erhalten. Zumindest ein peripherer Bereich wird mit Hilfe des Kernbereichs der Bilddaten bestimmt. Folglich ist es möglich, die Informationsmenge zu verringern, die bei der Erzeugung der vorerwähnten Codes zu verwenden ist, so daß die Codeerzeugung effektiv durchgeführt werden kann.

Gemäß der Erfindung sollen ein Bilddaten-Verarbeitungssystem und -Verfahren geschaffen werden, welche durch Verbessern des vorstehend beschriebenen Systems und Verfahrens erhalten werden. Diese Verbesserung betrifft eine Verbesserung in der Auflösung sich ergebender Bilder, d. h. eine Verbesserung in der Qualität von sich ergebenden Bildern. Die Verbesserung schließt ferner eine Reduzierung der Datenmenge ein, welche vorher in einem Speicher gespeichert werden sollen, wobei die Daten für die vorerwähnten Bildqualität-Verbesserungen verwendet werden. Die Verbesserung schließt auch eine Benutzungsmöglichkeit des relevanten Systems auf verschiedene Weise ein. Die Verbesserung schließt ferner eine Wirkungsgradverbesserung einer Bilddaten-Handhabung ein.

Gemäß der Erfindung ist dies bei einem Bilddaten-Verarbeitungssystem und einem -Verfahren durch die Merkmale in den Ansprüchen 1, 17, sowie 21 bis 26 erreicht. Vorteilhafte Weiterbildungen der Erfindung sind Gegenstand der auf einen der vorstehenden Ansprüche unmittelbar oder mittelbar rückbezogenen Ansprüche.

Um das vorstehend angeführte Ziel der Erfindung zu erreichen, werden in dem Bilddaten-Verarbeitungssystem und -Verfahren die Bilddaten wiederholt. Das heißt, in dem Signal der Bilddaten werden dieselben Wellenformen wiederholt. Folglich wird daß Bitmap, das aus den Bilddaten gemacht worden ist, beispielsweise dasjenige mit der doppelten Abmessung. Die doppelte Abmessung des Bitmap wird durch Verdoppeln jeder Linie erhalten, so daß jedes Bit in der zu den Liniensenkrechten Richtung verdoppelt wird. Jedes Paar derselben Bits wird verwendet, um die Zacken zu korrigieren, und das Verdoppeln der Bits bewirkt das Verdoppeln der sich ergebenden Bildauflösung und damit eine Verbesserung der sich ergebenden Bildqualität.

Ferner ist in dem System eine Zeitsteuersignale erzeugende Einrichtung vorgesehen. Folglich wird die

Zeitsteuerung der vorerwähnten Datenwiederholoperation synchron mit der zeitlichen Steuerung durchgeführt, welche durch die das Zeitsteuersignal erzeugende Einrichtung vorgegeben ist. Ferner kann die Anzahl N, wie oft die Bilddaten wiederholt werden, eingestellt werden. Somit kann die Bilddatenauflösung mit N multipliziert werden. Folglich kann die Bildqualität des sich ergebenden Bildes durch die Zacken-Korrekturoperation mit Hilfe der N-fachen Auflösungsbilddaten verbessert werden.

Ferner können die Bilddaten, welche durch die Wiederholoperation gerade erzeugt worden sind, mit Hilfe entsprechender Codes gezählt werden. Folglich können die gerade erzeugten Bilddaten richtig identifiziert werden.

Ferner wird die Korrekturart-Befehlsinformation (oder die Korrekturdaten, die zu verwenden sind, um die eingegebenen ursprünglichen Daten zu ersetzen), welche zum Durchführen der Zacken-Korrektur erforderlich ist, in der Menge verringert, ohne die signifikante Zacken-Korrekturfunktion zu verschlechtern, wodurch die Speicherkapazität verkleinert wird, die zum Speichern der Information erforderlich ist.

Ferner werden ein Tabellen- und ein Musterspeicher als die vorerwähnten Speicher verwendet, um die Korrekturart-Befehlsinformation zu speichern. Folglich kann die vorerwähnte Korrekturart-Befehlsinformation durch eine andere ersetzt werden, so daß verschiedene Korrekturmuster erhalten werden können. Darüber hinaus werden die Bilddaten, welche der Zacken-Korrekturoperation unterzogen worden sind, direkt als Daten verwendet, um die Laserdioden-Lichtemissionsenergie zu steuern. Folglich kann die Bilddaten-Verarbeitungsgeschwindigkeit verbessert werden.

Darüber hinaus wird eine entsprechende Software für eine Zentraleinheit (CPU) u.ä. verwendet, um ein endgültiges Bild (bezüglich Vergrößerung oder Verkleinerung) zu manipulieren und bei der Manipulations-Operation werden Bilddaten von denjenigen verwendet, die sich aus der unterschiedlichen (schwarz und weiß) Punktgrenzen-Liniensform-Erkennung ergeben, was aus dem Tabellspeicher und aus dem Musterspeicher erhalten worden ist.

Es ist auch möglich, die Zacken-Korrektur-Operation an den Bilddaten durchzuführen, die als ein Ergebnis zur Durchführung der gewünschten Bildmanipulation erhalten worden sind. Auch ist es möglich, sowohl die Zacken-Korrektur- als auch die Bildmanipulation-Operationen parallel durchzuführen.

Das Bilddaten-Verarbeitungssystem kann auf zwei Arten verwendet werden, indem entweder das Taktsignal, welches mit dem in dem System erzeugten Systemtakt synchronisiert ist, oder dasjenige verwendet werden kann, das von außen dem System zugeführt worden ist.

Durch Festsetzen einer Anzahl von Fenstern, um die entsprechenden Bereiche in dem Bitmap festzulegen, ist es ferner möglich, verschiedene Arten von Zacken-Korrektur-Operationen an den Bereichen beliebig durchzuführen. Folglich kann das endgültige Bild beigesteuert werden, und verschiedene Arten einer Bildmodifikation können an den Bilddaten durchgeführt werden, um so das gewünschte endgültige Bild zu verwirklichen. Durch Festlegen einer Fensterpriorität für die auf diese Weise festgelegte Anzahl von Fenstern können zwei Fenster bestimmt werden, so daß die entsprechenden Bereiche einander überdecken.

Ferner ist es möglich, verschiedene Arten eines Bild-

handhabungsprozesses zusätzlich einzustellen und zusammen mit einem solchen Zacken-Korrekturprozeß durchzuführen, und es ist möglich, für jedes Fenster festzulegen, ob solche Verarbeitungsarten durchgeführt werden oder nicht, oder welche Art eines Bildhandhabungsprozesses durchzuführen ist.

Somit führen das Bilddaten-Verarbeitungssystem und -Verfahren gemäß der Erfindung eine Zacken-Korrektur durch, indem Zacken aus sich ergebenden Bildern entfernt werden. Somit kann die Bildqualität in den sich ergebenden Bildern verbessert werden. Ferner erfordern das System und das Verfahren eine kleine Datensumme, welche vorher in einem Speicher gespeichert werden soll, wobei die Daten in der vorerwähnten Zacken-Korrektur-Operation notwendig sind. Ferner erfordern das System und das Verfahren eine kurze Zeitspanne für die Zacken-Korrektur-Operation. Darüber hinaus ist es auch möglich, eine Bilddaten-Auflösung zu verbessern, das System auf verschiedene Weise zu verwenden, und um ferner verschiedene Arten von Bildhandhabungsprozessen zusätzlich zu dem Zacken-Korrektur-Prozeß durchzuführen.

Nachfolgend wird die Erfindung anhand von bevorzugten Ausführungsformen unter Bezugnahme auf die anliegenden Zeichnungen im einzelnen erläutert. Es zeigen:

Fig. 1 ein Blockdiagramm eines Steuersystems in einem Laserdrucker in einer Ausführungsform der Erfindung zusammen mit einem Host-Computer;

Fig. 2 eine Schnittansicht eines mechanischen Aufbaus des Laserdruckers;

Fig. 3 eine perspektivische Ansicht einer Ausführungsform von wesentlichen Komponenten eines optischen Systems einer Schreibeinheit in dem in Fig. 2 dargestellten Aufbau;

Fig. 4 ein Blockdiagramm einer Ausführungsform einer Punktkorrektoreinheit der in Fig. 1 dargestellten Einrichtung;

Fig. 5 ein Blockdiagramm einer Ausführungsform eines FIFO-Speichers und einer Fenstereinheit in der in Fig. 4 dargestellten Einrichtung;

Fig. 6A bis 6I Zeitdiagramme, welche ein Beispiel von Operationen in dem in Fig. 5 dargestellten FIFO-Speicher 72 veranschaulichen;

Fig. 7A bis 7I Zeitdiagramme, welche ein weiteres Beispiel von Operationen in dem in Fig. 5 dargestellten FIFO-Speicher 72 veranschaulichen;

Fig. 8 ein Beispiel eines Abfragefensters, was durch die in Fig. 5 dargestellte Fenstereinheit realisiert ist;

Fig. 9 ein Blockdiagramm einer Ausführungsform einer in Fig. 4 dargestellten Muster-Erkennungseinheit und der relevanten Ausgangssignale;

Fig. 10 bis 13 Blockdiagramme von Ausführungsformen eines in Fig. 4 dargestellten Speicherblocks 75;

Fig. 14 zum Vergleich Speicherkapazitäten, welche für die jeweiligen in Fig. 10 bis 13 dargestellten Speicherblöcke erforderlich sind;

Fig. 15 ein Blockdiagramm einer Codeinformations-Schalteinrichtung, um einzugebende Code-Information als einen Teil einer Adresse in einem der in Fig. 11 bis 13 dargestellten Speicherblöcke zu schalten;

Fig. 16 ein Schaltungsdiagramm einer Ausführungsform der Codeinformations-Schalteinrichtung;

Fig. 17A bis 17F eine Zacken-Korrekturoperation, in welcher Korrekturdaten-Bildbruchstücke mit Hilfe der Codeinformations-Schalteinrichtung entsprechend angeordnet sind;

Fig. 18 bis 20 Blockdiagramme von Ausführungsfor-

men, in welchen jeweils eine Bilddaten-Handhabungseinrichtung in einer in Fig. 1 dargestellten Punktkorrekturseinheit vorgesehen ist;

Fig. 21 ein Blockdiagramm einer Ausführungsform, in welcher eine Fensterbereich-Einstelleinrichtung und eine Fensterbereich-Bilddaten-Verarbeitungseinrichtung in der Punktkorrekturseinheit vorgesehen sind;

Fig. 22 eine Operation in der in Fig. 21 dargestellten Ausführungsform;

Fig. 23 ein Blockdiagramm einer Ausführungsform, in welcher eine Fensterpriorität-Einstelleinrichtung in der in Fig. 21 dargestellten Punktkorrekturseinheit vorgesehen ist;

Fig. 24 eine Operation in der in Fig. 23 dargestellten Ausführungsform;

Fig. 25 ein Blockdiagramm einer Ausführungsform, in welcher eine Fensterbereich-Bilddaten-Verarbeitungsmodus-Einstelleinrichtung in der in Fig. 23 dargestellten Punktkorrekturseinheit vorgesehen ist;

Fig. 26A bis 26E eine Operation in der in Fig. 25 dargestellten Ausführung;

Fig. 27 ein Schaltungsdiagramm einer Ausführungsform einer Zeitsteuersignale erzeugenden Einrichtung, die in der in Fig. 4 dargestellten Steuersignal-Erzeugungseinrichtung vorgesehen ist;

Fig. 28A bis 28F eine Operation in der in Fig. 27 dargestellten Zeitsignale erzeugenden Einrichtung;

Fig. 29 ein Schaltungsdiagramm einer weiteren Ausführungsform der Zeitsignale erzeugenden Einrichtung;

Fig. 30A bis 30I eine Operation in der in Fig. 29 dargestellten ein Zeitsignal erzeugenden Einrichtung;

Fig. 31A bis 31D verschiedene Arten von Linienerkennungsmustern, um eine unter 45° verlaufende, schräge Linie in einem Kernbereich in einem in Fig. 8 dargestellten Fenster zu erkennen;

Fig. 32A bis 32G verschiedene Arten von Linienerkennungsmustern, um eine horizontale oder eine annähernd horizontale, schräge Linie in dem Kernbereich zu erkennen;

Fig. 33A bis 33G verschiedene Arten von Linienerkennungsmustern, um eine vertikale Linie oder eine annähernd vertikale schräge Linie in dem Kernbereich zu erkennen;

Fig. 34A bis 34D einen rechten, einen linken, einen oberen und einen unteren Bereich, die als periphere Bereiche für einen Kernbereich in einem in Fig. 8 dargestellten Fenster wirksam sind;

Fig. 35A bis 35C drei Untermengen für den rechten und linken Bereich;

Fig. 36A bis 36C drei Untermengen für den oberen und unteren Bereich;

Fig. 37 und 38 Beispiele, in welchen Untermengen als ein Ergebnis eines annähernd horizontalen und eines annähernd vertikalen Liniemusters ausgewählt werden, die zu erkennen sind;

Fig. 39 bis 41 Beispiele von Liniemustern, eines Musters, eines weiteren Musters, das sich aus jedem Bit ergibt, das um ein Bit nach rechts verschoben wird, und ein anderes Muster, das sich aus jedem Bit ergibt, das weiter um ein Bit nach rechts geschoben wird, welche in dem Fenster vorhanden sind, um Berechnungsbeispiele von Gradienten und Positionen mittels einer Gradienten-Berechnungseinheit und einer Positions-Berechnungseinheit zu erläutern, die in Fig. 9 dargestellt sind;

Fig. 42A und 42B ein Korrekturbeispiel, in welchem die in Fig. 4 dargestellte Punktkorrekturseinheit die Punktkorrektur für eine annähernd horizontale Linie und das entsprechende Laser-Pulsbreiten-Diagramm

durchführt;

Fig. 43A und 43B ein Korrekturbeispiel, in welchem die Punktkorrekturseinheit für eine annähernde vertikale Linie und das entsprechende Laser-Pulsphasen-Diagramm durchgeführt werden, und

Fig. 44A bis 44D Daten, die als ein Ergebnis erhalten worden sind, um Muster von relevanten Punkten (die zentralen Punkte in den Kernbereichen für die in Fig. 37 und 38 dargestellten Fälle mittels einer in Fig. 4 dargestellten Mustererkennungseinheit zu erkennen.

Nunmehr werden ein Laser-Printer/Drucker einer Ausführungsform eines Bilddaten-Verarbeitungssystems und -Verfahrens gemäß der Erfindung anhand von Fig. 1 beschrieben. Der Laser-Drucker/Printer 2 weist eine Steuereinheit 3, eine Antriebs-Ansteuereinheit 4, einen Druckerantrieb 5 und ein internes Interface 6 auf.

Der Laser-Drucker 2 erhält Druckdaten von einem Host-Computer 1 und die Steuereinheit 3 setzt die Druckdaten für jede Seite in Bitmap-Daten um. Die Steuereinheit 3 setzt dann die Bitmap-Daten in Videodaten um, welche Punktinformationen sind, welche einen Laser ansteuert. Die Steuereinheit 3 sendet die Videodaten über das interne Interface 6 an die Antriebs-Ansteuereinheit 4, so daß der Drucker 5 von Antriebs-Ansteuereinheit 4 gesteuert wird. Folglich erzeugt der Drucker 5 das relevante sichtbare Bild auf einem Blatt.

Eine Punktkorrekturseinheit 7 ist in dem internen Interface 6 vorgesehen, und die Einheit 7 verwendet das Bilddaten-Verarbeitungssystem und -verfahren gemäß der Erfindung. Die Einheit 7 führt einen Punkt, dessen Korrektur beschrieben wird, in den vorerwähnten Videodaten aus, die von der Steuereinheit 3 zugeführt worden sind. Somit wird die Bildqualität in dem gedruckten sichtbaren Bild verbessert.

Die Steuereinheit 3 enthält einen Haupt-Mikrocomputer 31 (welcher nachstehend als MPU bezeichnet wird) und einen ROM 32, welcher vorher von der MPU 31 verwendet Programme, konstante Daten, Text-Schriftartdaten usw. speichert. Die Steuereinheit 3 enthält ferner einen RAM 33 zum Speichern vorübergehender Daten, von Punkt muster-Daten usw., eine Ein/Ausgabe(I/O)-Einheit 34 zum Steuern der Daten Ein/Ausgabe und ein Bedienungsfeld 35, welches mit der MPU 31 über die I/O-Einheit 34 verbunden ist. Die vorstehenden Komponenten sind miteinander mittels Daten-, Adressen-, Steuer-Bussen, usw. verbunden.

Der Host-Computer (das Gerät) 1 und das interne Interface 6 einschließlich der Datenkorrekturseinheit sind über die I/O-Einheit 34 mit der MPU 31 verbunden. Die Antriebs-Ansteuereinheit 4 weist einen Sub-Mikrocomputer 41 (welcher als CPU nachstehend bezeichnet wird) und einen ROM 42 auf, welcher vorher von der CPU 41 verwendete Programme, konstante Daten usw. speichert. Die Antriebs-Ansteuereinheit 4 enthält ferner einen RAM 43 zum Speichern vorübergehender Daten und eine I/O-Einheit 44 zum Steuern einer Daten-Ein/Ausgabe. Die vorerwähnten Komponenten sind miteinander mittels Daten-, Adressen-, Steuer-Bussen usw. verbunden.

Die I/O-Einheit 44, die mit dem internen Interface 6 verbunden ist, gibt die Videodaten von der Steuereinheit 3 und auch die Zustände verschiedener Arten von Schaltern ein, die auf dem Bedienungsfeld 35 vorgesehen sind. Die I/O-Einheit 44 gibt ein Bildtaktsignal (WCLK) und Zustandssignale, wie ein Papierende-Signal, an die Steuereinheit 3 ab.

Ferner sind auch die I/O-Einheit 44 mit einer Schreib-

einheit 2, eine eine Druckfolge durchführende Gerätegruppe 27 und verschiedenen Arten von Sensoren 28 einschließlich eines Synchron-Sensors verbunden, was noch beschrieben wird. Diese Komponenten sind in dem Drucker 5 vorgesehen.

Die Steuereinheit 3 erhält Befehle, wie Druckbefehle, und die Druckdaten, wie Textdaten, graphische Bilddaten usw. von dem Host-Computer 1. Die Steuereinheit 3 bereitet die erhaltenen Druckdaten auf und die Steuereinheit erhält die entsprechenden Punktmuster, wenn die Druckdaten die Textdaten enthalten. Solche Punktmuster können als die Textschriftarten erhalten werden, die in dem ROM 32 gespeichert sind und werden verwendet, um die entsprechenden Texte auszudrucken. Die Steuereinheit 3 erzeugt die entsprechenden Bitmap-Daten, die aus Texten und graphischen Bildern bestehen (von welchem zwei der Einfachheit halber nachstehend als Bilder bezeichnet werden). Die Steuereinheit 3 speichert dann die Bitmap-Daten in einem Video-RAM-Bereich in dem RAM 33 in einem Bitmap-Format für jede Seite.

Die Steuereinheit 3 erhält ein Bereit-Signal und ein Bildtaktsignals WCLK, das von der Ansteuereinheit 4 zugeführt worden ist. Folglich gibt die Steuereinheit 3 Bitmap-Daten (Punktmusterdaten) ab, welche in dem vorerwähnten Video-RAM-Bereich in dem RAM 33 gespeichert sind, wie vorstehend erwähnt ist. Die abgegebenen Bilddaten wirken als die Videodaten und werden der Ansteuereinheit 4 über das interne Interface 6 synchron mit den Taktten des Bildtaktsignals WCLK zugeführt. Die Videodaten sind Daten, welche einer Punkt-korrekturoperation, welche noch beschrieben wird, mittels der Punktkorrekturereinheit 7 in dem internen Interface 6 unterzogen werden.

Das Bedienungsfeld 35 hat verschiedene Schalter und Anzeigen, welche in der Figur nicht dargestellt sind. Eine Bedienungsperson benutzt solche Daten, um verschiedene Befehle und Daten an dem Laserdrucker 2 zu spezifizieren. Das Bedienungsfeld 35 überträgt folglich die so erhaltene Information an die Ansteuereinheit und zeigt verschiedene Zustände in dem Drucker 2 über die Anzeigen an.

Die Ansteuereinheit 4 verwendet die Videodaten, welche über das Interface 6 von der Steuereinheit 3 zugeführt worden sind. Folglich steuert die Ansteuereinheit 4 die Schreibeinheit 6, die eine Druckfolge durchführende Gerätegruppe 27 usw. Die Gerätegruppe 27 enthält einen Lader und eine Entwicklungseinheit. Die Ansteuereinheit 4 gibt die Videodaten über das interne Interface 6 ein und gibt die Videodaten an die Schreibeinheit 26 ab. Die Ansteuereinheit 4 gibt ferner Signale, welche Zustände verschiedener Teile in der Einrichtung anzeigen, von verschiedenen Arten von Sensoren 28 ein. Die Ansteuereinheit 4 verarbeitet folglich die auf diese Weise eingegebenen Signale und gibt die Zustandssignale, wie ein Fehlerzustandsignal sowie ein Papierende-Zustandssignal und andere Information, die notwendigerweise zugeführt werden muß, über das Interface 6 an die Steuereinheit 3 ab.

Anhand von Fig. 2 wird ein mechanischer Aufbau des Druckers 5 in dem Laserdrucker 2 beschrieben. Der Laserdrucker 2 führt Papierblätter 11 entweder von der oberen oder der unteren Papierzuführ-Kassette 12a oder 12b zu. Ein Blatt 11 soll beispielsweise von dem Blattstapel 11a in der oberen Kassette 10a über eine Papierzuführrolle 12 zugeführt werden. Das Blatt 11 wird in die Bildübertragungsposition auf einer photoempfindlichen Trommel 11 befördert, wobei die zeitlich

gesteuerte Bewegung des Blattes 11 durch ein Ausricht-Rollenpaar 13 gesteuert wird.

Der Hauptmotor 14 dreht die Trommel 15 in einer durch einen Pfeil in Fig. 2 angezeigten Richtung. Die Oberfläche der Trommel wird mittels des Laders 16 geladen, und dann wird ein elektrostatisches, latentes Bild auf die Oberfläche der Trommel 15 erzeugt, wobei die Oberfläche der Trommel 15 durch einen Lichtpunkt entsprechend abgetastet wird. Der Lichtpunkt wird von der Schreibeinheit 26 geliefert und ist pulsbreiten-(PW-)moduliert.

Das auf der Oberfläche der Trommel 11 erzeugte, elektrostatische, latente Bild wird mit Hilfe von Toner, der auf die Trommeloberfläche aufgebracht ist, in ein sichtbares Bild umgewandelt. Das sichtbare Tonerbild wird dann mittels eines Übertragungsladers 18 an das Blatt 11 übertragen, welches dann von der Trommel 11 entfernt und mittels eines Förderbandes 19 in eine Fixiereinheit 20 befördert wird. Eine Andrückrolle 20a in der Einheit 20 drückt das Blatt 1 an eine Fixierrolle 20b in der Einheit 20 durch den Druck und die Temperatur der Fixierrolle 20b wird dann das auf dem Blatt 1 vorhandene Tonerbild auf dem Blatt 11 fixiert.

Das Blatt wird dann aus der Einheit 20 ausgetragen und mittels einer Austragrolle 21 auf eine Austragablage 22 befördert, die an einer Seite des Druckers 2 vorgesehen ist. Toner, der ungenutzt auf der Oberfläche der Trommel 15 verblieben ist, wird mittels einer Reinigungseinheit 23 entfernt und gesammelt. Der Laserdrucker 2 hat in dem oberen Teil in dem Drucker 2 eine Vielzahl gedruckter Schaltungen 24, wie in Fig. 2 dargestellt ist. Die gedruckten Schaltungen verkörpern die vorstehend beschriebene Steuereinheit 3, die Antriebsansteuereinheit 4 und das interne Interface 6.

Anhand von Fig. 3 wird nunmehr eine Ausführungsform der Schreibeinheit 26 beschrieben. Die Einheit 26 weist eine Laserdioden-Einheit 50 (die nachstehend als eine LD-Einheit bezeichnet wird), eine erste Zylinderlinse 51, einen ersten Spiegel 52, eine bilderzeugende Linsenanordnung 53, einen rotierenden Polarisator mit einem scheibenförmigen Motor 54 und einen Polygon-Spiegel 55, der von dem Motor 54 einer in Fig. 3 angezeigten Richtung A gedreht wird, einen zweiten Spiegel 57, eine zweite Zylinderlinse 58, einen dritten Spiegel 60, eine Licht-konvergierende Linse 61, einer Zylinderlinse und einen Synchron-Sensor 62 eines photoempfindlichen Elements auf. Die LD-Einheit 50 hat eine Laserdiode (welche als LD abgekürzt wird) und eine Kollimatorlinse, die in der Einheit 50 integriert ist. Die Kollimatorlinse formt einen von der LED abgegebenen, divergenten Strahl in einen parallelen Strahl um.

Die erste Zylinderlinse 51 ändert in der Unterabtastrichtung auf der Oberfläche der Trommel 15 die Form des von der LD-Einheit 50 gelieferten, parallelen Strahl. Die Bilderzeugungslinse 53 formt einen von dem ersten Spiegel 52 reflektierten, parallelen Strahl in einen konvergenten Strahl um, und der von der Bilderzeugungslinse 53 abgegebene Strahl trifft auf eine der Spiegelflächen 55a des Polygonspiegels 55.

Jede der Spiegelflächen 55a des Polygonspiegels 55 hat eine gekrümmte Form, so daß der Polygonspiegel 55 ein R-Polygonspiegel ist. Folglich kann eine fO-Linse entfallen, welche üblicherweise zwischen dem Polygonspiegel 55 und dem zweiten Spiegel 57 angeordnet ist. Ein Lichtstrahl trifft auf den rotierenden Polarisator 56, nachdem er in einen konvergierenden Strahl umgewandelt ist.

Der zweite Spiegel 57 reflektiert den Strahl, welcher

von dem rotierenden Polarisator 56 reflektiert und polarisiert worden ist. Der reflektierte Strahl ist ein abstrahlender Strahl und trifft über die zweite Zylinderlinse 58 auf die photoempfindliche Trommel 15 auf. Der Abtaststrahl konvergiert folglich, um einen scharfen Punkt auf der Oberfläche der Trommel auf einer Hauptabtastzeile 15a zu erzeugen.

Der dritte Spiegel 60 ist in einer Position außerhalb der Zone auf der Trommel 15 angeordnet, in welcher Zone der Lichtstrahl abtastet, welcher von dem rotierenden Polarisator 56 reflektiert worden ist. Der dritte Spiegel 60 reflektiert auffallendes Licht auf den Synchron-Sender 62. Der von dem dritten Spiegel 60 reflektierte Strahl wird dann durch die Konvergenzlinse 61 so konvergiert, daß der sich ergebene Strahl von dem Synchronsensor 62, der ein photoempfindliches Element, wie eine Photodiode aufweist, empfangen wird und ihn in ein synchrones Signal umsetzt, das zu verwenden ist, um die Abtaststartposition festzulegen.

Anhand von Fig. 4 und 5 wird nunmehr die Punktkorrekturseinheit 7 beschrieben. Die Punktkorrekturseinheit 7 weist auf einen Parallel/Seriell-(P/W-)Umsetzer 71, einen FIFO-Speicher 72, eine Fenstereinheit 73, eine Mustererkennungseinheit 74, einen Speicherblock 75, eine Videodaten-Abgabeeinheit 76 und eine Zeitsteuerseinheit 77, welche die vorstehend angeführten Komponenten steuert, so daß sie synchron miteinander arbeiten.

Der P/S-Umsetzer ist vorgesehen, um die Videodaten, welche von der in Fig. 1 dargestellten Steuereinheit 3 zugeführt worden sind, in die seriellen Einbit-Daten umzusetzen, wenn die Videodaten ursprünglich 8 Bit Paralleldaten sind. Die auf diese Weise umgesetzten seriellen Einbit-Daten werden dann dem FIFO-Speicher 72 zugeführt. Folglich ist der P/S-Umsetzer 71 keine Komponente, welche für die Punktkorrektur-Operation wesentlich ist, und er ist nicht notwendig, wenn die von der Steuereinheit zugeführten Videodaten serielle Daten sind.

Der FIFO-Speicher 72 weist Zeilenpuffer 72a bis 72g auf, die über einen Multiplexer 721, welcher auch den FIFO-Speicher 72, wie in Fig. 5 dargestellt, enthält, in Reihe miteinander geschaltet sind. Die Zeilenpuffer 72a bis 72g speichern die Videodaten einer Anzahl von Zeilen (7 Zeilen dieser Ausführungsform) in einem Bild, welche Videodaten von der Steuereinheit 3 zugeführt worden sind.

Wenn eine noch zu beschreibende Zeitsteuereinheit 77 den Wert "0" als ein DATA-SEL-Signal liefert, wählt der Multiplexer 721 in Fig. 5 dargestellte A-Eingangssignale aus. D.h. der Multiplexer gibt die von der Steuereinheit 3 über den P/S-Umsetzer 71 zugeführten Videodaten, die 72a-, 72b-, 72c-, 72d-, 72e- und 72f-Ausgabedaten ein. Dann gibt der Multiplexer 721 die auf diese Weise eingegebenen Daten an die entsprechenden Zeilenpuffer 72a bis 72g ab. Wenn die Zeitsteuereinheit 77 den Wert "1" als das DATA-SEL-Signal liefert, wählt der Multiplexer B-Eingabedaten aus. D. h. der Multiplexer gibt die 72a- bis 72g-Ausgabedaten ein. Dann gibt der Multiplexer 721 die auf diese Weise eingegebenen Daten an die entsprechenden Zeilenpuffer 72a bis 72g ab.

Folglich liefern die Zeilenpuffer 72a bis 72g des FIFO-Speichers 72 in Fig. 6C bis 6I und 7C bis 7I dargestellten Daten entsprechend den Videodaten, wie sie in Fig. 6A und 7A dargestellt sind ab, die entsprechend den in Fig. 6B und 7B dargestellten DATA-SEL-Signale zugeführt werden. Der FIFO-Speicher 72 hat eine Funktion, daß nur ein Schreibbefehl an jeden der Zeilenpuffer 72a

bis 72g während eines Zeitabschnitts abgegeben wird, in welchem das DATA-SEL-Signal den Wert "0" hat. Folglich schreiben die Zeilenpuffer 72a bis 72g nur die ihnen zugeführten Daten ein. Sowohl der Schreib- als auch ein Lesebefehl werden an jeden der Puffer 72a bis 72g während eines Zeitabschnitts angeführt, in welchem das DATA-SEL-Signal den Wert "1" hat. Folglich schreibt jeder der Zeilenpuffer 72a bis 72g das Bit der zugeführten Daten ein und liest das Datenbit, das in ihn gespeichert worden ist, entsprechend der FIFO-Weise; die Einbit-Schreib- sowie die Einbit-Leseoperation werden gleichzeitig ausgeführt. Folglich wirkt der FIFO-Speicher 72 als eine Bilddaten erzeugende Einrichtung.

Wie in Fig. 5 dargestellt, weist die Fenstereinheit 73 sieben 11 Bit-Schieberegister 73a bis 73g für die sieben Zeilen von Bilddaten, welche durch die sieben Zeilenpuffer 72a bis 72g abgegeben werden sind. Die 11 Bit-Bereiche jedes Schieberegisters sind in Reihe mit einem der vorerwähnten Zeilenpuffer verbunden. Die sieben Schieberegister 73a bis 73g entsprechen einem Fenster (Abfragefenster), welches verwendet wird, um ein Muster in dem durch eingegebene Videodaten dargestellten Bild zu führen. Fig. 8 zeigt ein Beispiel einer Form eines solchen Fensters.

In der Fenstereinheit 73 dient ein in Fig. 5 dargestellter Bitbereich 73dx der mittlere Bereich des mittleren Schieberegisters 73d, um einen relevanten Punkt (d. h. einem Punkt, welcher augenblicklich verarbeitet wird) unter den durch das Fenster festgelegten Punkten zu speichern. Von den Bitpositionen der sieben Schieberegister 73a bis 73g können die durch gestrichelte Linien in Fig. 5 dargestellten Bitpositionen weggelassen werden, wenn die in Fig. 8 dargestellte Fensterform verwendet wird. D.h. jedes der Schieberegister 73a und 73g sollte sieben Bitpositionen und jedes der Schieberegister 73b und 73f sollte 8 Bitpositionen haben.

Bits von eingegebenen Bilddaten bewegen sich nacheinander Bit für Bit durch die Zeilenpuffer 72a bis 72g und die Schieberegister 73a bis 73g. Durch diese Bewegung ändert sich ein Bit, das dem vorerwähnten relevanten Punkt entspricht, in ein anderes Bit. Schließlich sind alle Bits von eingegebenen Videodaten somit der relevante Punkt. Folglich kann das jeweilige Fenster 73 nacheinander für alle Bits erhalten werden, wobei jedes Bit dem jeweils einen der relevanten Punkte der Fenster von eingegebenen Videodaten entspricht.

Die Mustererkennungseinheit 74 erkennt die charakteristischen Daten einer Form, die in jedem Fenster 73 erscheint mit Hilfe der entsprechenden, von der Fenstereinheit 73 erhaltenen Punktinformation. Eine derartige Form ist insbesondere eine Form der Grenze zwischen einem Schwarzpunkt- und einem Weißpunkt-Bereich, welcher an und um den relevanten Punkt in dem Fenster 73 vorhanden ist. Die Einheit 74 wandelt nach Erkennen solcher charakteristischer Daten die charakteristischen Daten in ein vorherbestimmtes Format von Codeinformation um, welche dann von der Einheit 74 abgegeben wird. Eine derartige Codeinformation wird als ein Adressencode verwendet, welcher die Adresse in dem in Fig. 4 dargestellten Speicherblock 75 anzeigt.

Anhand von Fig. 9 wird nunmehr ein innerer Aufbau der Mustererkennungseinheit 74 in Verbindung mit Fenstern 73, wie sie vorstehend erwähnt sind, beschrieben. Wie aus Fig. 34A bis 34D zu ersehen, besteht jedes Fenster 73 aus einem Kernbereich 73C, welcher in der Mitte des Fensters angeordnet ist und aus (3 x 3)-Punkten gebildet ist, aus einem oberen Bereich 73U, einem unteren Bereich 73D, einem linken Bereich 73L und

einem rechten Bereich 73R. Die Bereiche 73U, 73D, 73L und 73R sind, wie in den Figuren dargestellt, um den Kernbereich C herum angeordnet, sie werden später beschrieben.

Die Mustererkennungseinheit 74 enthält eine Kernbereich-Erkennungseinheit 47, eine Peripherbereich-Erkennungseinheit 742, Multiplexer 743, 744, eine Gradienten-Berechnungseinheit 745, eine Positions-Berechnungseinheit 776, eine Bestimmungseinheit 747 und ein Verknüpfungsglied 748. Die Peripherbereich-Erkennungseinheit 742 enthält eine den oberen Bereich erkennende Einheit 742U, eine den rechten Bereich erkennende Einheit 742R, eine den unteren Bereich erkennende Einheit 742D und eine den linken Bereich erkennende Einheit 742L.

Jede Einheit hat dieselbe Funktion wie die entsprechende Einheit, welche in der vorerwähnten offengelegten japanischen Patentanmeldung Nr. 5-207 282 beschrieben ist und hier später noch beschrieben wird.

Ausführungsformen von Blockformation und die Arbeitsweise des Speicherblocks 75 gemäß der Erfindung, was in Fig. 4 dargestellt ist, d. h. eine Korrekturdaten-Abgabeeinrichtung wird nunmehr anhand von Fig. 10 bis 13 beschrieben.

Nunmehr wird die in Fig. 10 dargestellte Ausführungsform beschrieben, welche dieselbe ist die in der vorerwähnten Anmeldung Nr. 5-207 282. Der Speicherblock 75 enthält nur einen Musterspeicher 752. Codeinformation die, wie vorstehend erwähnt, von der Mustererkennungseinheit 74 zugeführt worden ist, wird als die entsprechende Adresse in dem Musterspeicher 752 verwendet. Folglich werden die vorher gespeicherten Korrekturdaten aus dem Musterspeicher 752 ausgelesen, wenn die Daten an der durch die Codeinformation angezeigten Adresse gespeichert ist. In dem Musterspeicher gespeicherte Korrekturdaten stellen Bildteile (Punktmuster) dar, welche verwendet werden, um Bildteile zu ersetzen, welche durch eingegebene Bilddaten dargestellt sind. Folglich werden die eingegebenen Bilddaten korrigiert. Die auf diese Weise gelesenen Korrekturdaten werden dann als Videodaten zum Ansteuern eines Lasers verwendet. Somit werden die Korrekturdaten das Punktmuster, welche sich aus der Korrektur ergibt, die durchgeführt worden ist.

Nunmehr wird die in Fig. 11 dargestellte Ausführungsform beschrieben. Auch in dieser Ausführungsform enthält der Blockspeicher 75 nur einen Musterspeicher 752. Die Unterschiede zwischen den Ausführungsformen der Fig. 10 und 11 werden nunmehr beschrieben. Korrekturdaten werden aus dem Musterspeicher 752 als sowohl Codeinformation, die von der Mustererkennungseinheit 74 zugeführt worden ist als auch andere Codeinformation, die von einer Zeitsteuersignal erzeugenden Einrichtung zugeführt worden ist, welche in der in Fig. 4 dargestellten Zeitsteuereinheit 77 vorgesehen ist. Die Codeinformation wird als die Adresse in dem Musterspeicher 752 verwendet. Die Codeinformation, welche von der Zeitsteuereinheit 77 zugeführt worden ist, zeigt eine wiederholte Reihenfolge von relevanten Daten an. Eine solche wiederholte Reihenfolge von relevanten Daten wird nunmehr beschrieben.

In der Ausführungsform der Erfindung wird jede Gruppe von Videodaten, welche von der in Fig. 1 dargestellten Steuereinheit 3 der Punktkorrekturteinheit 7 zugeführt worden ist, über die vorstehend beschriebene Funktion des FIFO-Speichers 72 wiederholt. Die Wiederholoperation wird während eines Zeitabschnitts durchgeführt, während welchem das in Fig. 6B und 7B

dargestellte DATA-SEL-Signal den hohen Pegelwert hat. In dem in Fig. 6A bis 6I dargestellten Fall wird jede Gruppe von Videodaten einmal wiederholt, und in dem in Fig. 7A bis 7I dargestellten Fall wird jede Gruppe von Videodaten dreimal wiederholt. Insbesondere in einem Fall, wie er in Fig. 7A bis 7I dargestellt ist, bei welchem eine Wiederholung mehrmals durchgeführt wird, werden die wiederholten Gruppen (welche als Kopien bezeichnet werden) von Videodaten identifiziert, in welcher Reihenfolge jede Kopie wiederholt worden ist. In dem in Fig. 7A bis 7I dargestellten Fall, hat jeder Datensatz drei Kopien, eine Kopie erster Ordnung (die als erste von dem ursprünglichen Datensatz erhalten worden ist), eine Kopie zweiter Ordnung (die als zweite von dem ursprünglichen Datensatz erhalten worden ist) und eine Kopie dritter Ordnung (die als dritte von dem ursprünglichen Datensatz erhalten worden ist). Die vorerwähnte Codeinformation, welche von der Zeitsteuereinheit 77 zugeführt worden ist, zeigt eine derartige Ordnung der jeweiligen Kopiedaten an. Diese Codeinformation wird durch in Fig. 11 dargestellte 4 Bit-Daten A12 bis A15 dargestellt. Die von der Mustererkennungseinheit 74 zugeführte Codeinformation ist durch 12 Bit Daten dargestellt. Folglich werden insgesamt 16 Bit Daten als eine Adresse in dem in Fig. 11 dargestellten Musterspeicher 752 verwendet.

Vorteile, welche durch die Ausführungsform der Fig. 11 im Vergleich zu der Ausführungsform der Fig. 10 erhalten worden sind, werden nunmehr beschrieben. Als Ergebnis einer Datenwiederholung, wie sie vorstehend beschrieben ist, enthalten Daten, welche der Mustererkennungseinheit 74 zugeführt worden sind, identische Datensätze für jeden ursprünglichen Datensatz, und die entsprechenden Sätze von Codeinformationen werden dem Musterspeicher 752 durch die Einheit 74 als 12 Bit Daten zugeführt. 4 Bit Daten in wiederholter Reihenfolge, wie vorstehend beschrieben ist, werden auch dem Musterspeicher durch die Zeitsteuereinheit 77 zusätzlich zu den vorerwähnten 12 Bit Daten zugeführt und somit können verschiedene Sätze von Korrekturdaten in dem Musterspeicher 752 für solche identische Sätze von eingegebenen Daten mit verschiedenen wiederholten Reihenfolgen gespeichert werden. In dieser Ausführungsform gemäß der Erfindung erzeugt die vorstehend beschriebene Wiederholoperation viele identische Sätze von Bilddaten für jede Zeile des durch eingegebene Bilddaten dargestellten, ursprünglichen Bildes. Die Mustererkennungseinheit 74 erkennt die Zeilenformen, die in einem Bitmap-Format durch viele identische Sätze von Bilddaten dargestellt sind. Folglich sind die auf diese Weise erhaltenen Codeinformationsätze identisch. Die identischen Codeinformationsätze können durch die 4 Bit in wiederholter Reihenfolge vorliegende Daten A12 bis A15 identifiziert werden. Praktische Beispiele einer Korrekturoperation mit einer solchen in wiederholter Reihenfolge wiedergegebenen Identifizierung wird später anhand von Fig. 17A bis 17F beschrieben.

Anhand von Fig. 12 wird eine Ausführungsform beschrieben, bei welcher der Speicherblock 75 einen Tabellenspeicher 751 zusätzlich zu dem Musterspeicher 752 enthält. 12 Bit Codeinformation, welche von der Mustererkennungseinheit 74 zugeführt worden ist, und 4 Bit Codeinformation A12 bis A15, wie sie vorstehend beschrieben ist, werden dem Tabellenspeicher 751 zugeführt. Andere Codeinformation, welche die Adressen in dem Musterspeicher 752 anzeigt, wird in dem Tabellenspeicher 751 gespeichert. Der entsprechende Satz

Codeinformation wird aus dem Tabellenspeicher 751 entsprechend den vorstehend wiedergegebenen 12 Bit Daten und 4 Bit Daten gelesen. Dann wird gemäß der Adresse in dem Musterspeicher 752, welche durch den ausgelesenen Satz Codedaten spezifiziert worden ist, der entsprechende Satz Korrekturdaten aus dem Musterspeicher 752 gelesen.

Eine Formation, wie sie beispielsweise in Fig. 12 dargestellt ist, ist unter einer Annahme effektiv, welche in vielen Fällen auf dem Gebiet der Bilderzeugungs-Technologie angewendet werden kann, und sie wird nunmehr beschrieben. In der Ausführungsform der Fig. 11 hat der Musterspeicher 752 eine Anzahl von Korrekturdatensätzen, und diese Anzahl ist dieselbe wie die Anzahl von möglichen verschiedenen Codeinformations-sätzen, welche von der Mustererkennungseinheit 74 zu führt werden können. Jedoch müssen eine Anzahl von Korrekturdatensätzen, d.h. eine Anzahl von Punktmustern, die zu verwenden sind, um ursprüngliche Bildteile zu ersetzen, nicht notwendigerweise dieselben sein, wie die angeforderte Anzahl von Codeinformationssätzen (Bildteil-Mustern, welche durch eingegebene Bilddaten dargestellt sind), welche von der Mustererkennungseinheit 74 zugeführt werden können. D.h. viele mögliche verschiedene Bildteil-Muster von eingegebenen Daten müssen durch ein gemeinsames Punktmuster zum Zwecke einer Zacken-Korrektur ersetzt werden, wie vorstehend beschrieben worden ist. Dies ist die vorstehend angegebene Annahme und ist eine Tatsache unter der Voraussetzung, daß eine wirksame und effektive Zacken-Korrektur versucht wird. Wenn die vorstehende Annahme bei der Ausführungsform der Fig. 11 angewendet wird, hat der Musterspeicher 752 viele Sätze identischer Datensätze, d.h. viele Sätze von identischen Datenmustern. Die Ausführungsform der Fig. 11 kann solche verdoppelten (oder mehrfach wiederholten) Punktmuster aus dem Musterspeicher 752 entfernen. Folglich kann die Gesamtspeicherkapazität des Speicherblocks 75 verkleinert werden, obwohl der Speicherblock 75 zwei Speicher 751 und 752 hat, ohne dadurch eine wesentliche Verschlechterung bei einer Zacken-Korrekturvornahme zu bewirken.

Durch ein Entfernen von verdoppelten Punktmustern in dem Speicherhuster 752 wird eine Anzahl Bits (Bitbreite) verringert, welche Codeinformation darstellen, welche dem Musterspeicher 752 zuzuführen ist. Es wird eine 7 Bit Codeinformation verwendet, um eine Korrekturinformation aus dem Musterspeicher 752 in den Ausführungsformen der Fig. 12 und 13 zu erhalten, während eine 12 Bit Codeinformation für denselben Zweck in den Ausführungsformen der Fig. 10 und 11 verwendet wird.

Ferner hat die Ausführungsform der Fig. 12 dieselben Vorteile wie diejenigen, die bei der Ausführungsform der Fig. 11 erhalten worden sind in dem in wiederholter Reihenfolge wiedergegebene Information verwendet wird, wie vorstehend beschrieben ist. Eine weitere Ausführungsform wird anhand von Fig. 13 beschrieben. Es werden Unterschiede zwischen den Ausführungsformen der Fig. 12 und 13 beschrieben. Die Codeinformation wird aus dem Tabellenspeicher 751 gelesen, welche Information vorher an der Adresse gespeichert ist, welche durch von der Mustererkennungseinheit 74 zugeführte Codeinformation angezeigt worden ist. Dann wird die Korrekturinformation aus dem Musterspeicher 752 gelesen, welche vorher an der Adresse gespeichert ist, welche sowohl durch aus dem Tabellenspeicher 751 ausgelesene Codeinformation als auch durch Codeinformati-

on A12 bis A15 angezeigt worden ist, wie vorstehend beschrieben ist.

Aus einem Grund, welcher demjenigen in der Ausführungsform der Fig. 12 entspricht, ist auch die Ausführungsform der Fig. 13 wirksam, um die Gesamtspeicherkapazität zu verringern, welche für den Speicherblock 75 erforderlich ist. Bei der Ausführungsform der Fig. 14 kann die Gesamtspeicherkapazität verringert werden, wenn die Anzahl an Bildteil-Mustern, welche unter Benutzung des Tabellenspeichers 751 behandelt werden kann, merklich größer ist als die Anzahl verschiedener Punktmuster, die als Korrekturinformation in dem Musterspeicher 752 gespeichert ist. Durch Hinzufügen von 4 Bit Codeinformation A12 bis A15 zu Information, welche Adressen in einem Speicher anzeigen, wird die Kapazität des Speichers mit  $2^4$  multipliziert. Wenn die Anzahl Muster, welche von dem Tabellenspeicher behandelt werden kann, erheblich größer ist als die Anzahl verschiedener Muster, die in dem Musterspeicher gespeichert sind, ist die Speicherkapazität des Tabellenspeichers 751, bevor sie mit  $2^4$  multipliziert wird, größer als diejenige des Musterspeichers 752 bevor diese mit  $2^4$  multipliziert wird. In einem solchen Fall ist es zum Verringern der Gesamtspeicherkapazität wirksamer, die Speicherkapazität des Musterspeichers statt diejenige des Tabellenspeichers mit  $2^4$  zu multiplizieren. Folglich wird bei der Ausführungsform der Fig. 14 die Gesamtspeicherkapazität effektiv verringert. (Siehe den in Fig. 14 dargestellten Vergleich.) Obwohl ein derartiger Speicher-Formationsweg wirksam ist, um die Gesamtspeicherkapazität zu verringern, wird dadurch keine Verschlechterung einer Zacken-Korrektur bewirkt. Ferner weist die Ausführungsform der Fig. 13 dieselben Vorteile auf, wie diejenigen, welche bei der Ausführungsform der Fig. 11 erhalten worden sind, indem eine durch eine wiederholte Reihenfolge gebildete Information verwendet wird, wie vorstehend beschrieben ist.

Fig. 14 zeigt einen Vergleich einer erforderlichen Speicherkapazität bei den Ausführungsformen der Fig. 10 bis 14. Jeder Satz Korrekturinformation, wie sie vorstehend beschrieben ist, um letztendlich aus dem Musterspeicher 752 ausgelesen zu werden, ist in jeder Ausführungsform eine 10 Bit Information (d.h. die Bitbreite ist 10). Die Anzahl an Bildteil-Mustern, welche von der Bildererkennungseinheit 74 zugeführt und von dem Blockspeicher 75 behandelt werden kann, beträgt ferner 4096 Bits in jeder Ausführungsform, und zwar deswegen, da der Codeinformationssatz, welcher von der Mustererkennungseinheit 74 zugeführt worden ist, eine 12 Bit Information ist ( $2^{12} = 4096$ ). Folglich beträgt in der Ausführungsform der Fig. 10 die Gesamtspeicherkapazität 40 960 Bit, da die vorerwähnten 4096 (Bits) und die 10 (Bitbreite einer Korrekturinformation) miteinander multipliziert werden. In der Ausführungsform der Fig. 11 ist durch das Hinzufügen von Codeinformation A12 bis A15 zu Information, welche als Adressen in dem Musterspeicher 752 zu verwenden ist, die Gesamtspeicherkapazität 655 360 Bits, da die 40 960 und  $2^4$  der Information A12 bis A15 multipliziert werden.

In der Ausführungsform der Fig. 12 ist die Codeinformation, welche als Adressen in dem Musterspeicher 752 zu verwenden sind, im Unterschied zu der 10 Bit Information, die in den Ausführungsformen der Fig. 10 und 11 verwendet ist, eine 7 Bit Information. Der Grund für diese Bitbreiten-Reduzierung wird nunmehr beschrieben. Die Anzahl Punktmuster, welche durch die in dem Musterspeicher gespeicherte Korrekturinformation

dargestellt sind, wird verringert. Dies ist möglich, da, wie vorstehend beschrieben, doppelte Punkt muster aus dem Musterspeicher entfernt werden. Folglich wird die Anzahl der gespeicherten Punkt muster von den vorstehend angeführten 4096 um 128 (= 2<sup>7</sup>) verringert. Folglich beträgt die Speicherkapazität des Musterspeichers in der Ausführungsform der Fig. 12 1280 Bits, wie in Fig. 14 dargestellt ist, indem die vorerwähnten 128 und die Korrekturinformations-Bitbreite 10 miteinander multipliziert werden. Die Kapazität von 458 752 Bits des Tabellenspeichers 751 in der Ausführungsform der Fig. 12 wird erhalten, indem die vorerwähnten 7 (die Bitbreite der Codeinformation, welche an dem Musterspeicher 752 als dessen Adressen zugeführt sind) und 2<sup>(12+4)</sup> miteinander multipliziert werden. Der Exponent (12 + 4) wird durch die Bitbreiten der von der Mustererkennungseinheit 74 geführten Codeinformation und der Information A12 bis A15 erhalten.

In der Ausführungsform der Fig. 13 wird die Kapazität von 20 480 Bits des Musterspeichers 752 dadurch erhalten, daß die 1280 Bits des Speichers 752 in der Ausführungsform der Fig. 12 und 2<sup>4</sup> multipliziert werden. Die Kapazität von 28 672 Bits des Tabellenspeichers 751 wird dadurch erhalten, daß die 458 752 Bits des Speichers 751 in der Ausführungsform der Fig. 12 durch dieselben 2<sup>4</sup> geteilt werden. Der Exponent 4 von 2<sup>4</sup> ist die Bitbreite der Information A12 bis A15. Diese Kapazitätsänderung führt dazu, daß der Speicher, in welchem sich die Information A12 bis A15 befindet, von dem Tabellenspeicher 751 in den Musterspeicher 752 geändert wird. Folglich kann die Gesamtspeicherkapazität in der Ausführungsform der Fig. 13 verringert werden, obwohl die Zacken-Korrektur im wesentlichen dieselbe ist wie bei den Ausführungsformen der Fig. 11 bis 13. Jede der Ausführungsformen der Fig. 10 bis 13 kann in der Ausführungsform gemäß der Erfindung verwendet werden.

Eine weitere in Fig. 15 und 16 dargestellte Ausführungsform der Erfindung wird nunmehr beschrieben. Die Ausführungsform der Fig. 15 hat eine Codeinformations-Schalteinrichtung 90, um die Information, welche dem Speicherblock 75 zuzuführen ist, von der Information A12 bis A15 auf eine andere Information RA12 bis RA15 und umgekehrt zu schalten. Dieser Schaltvorgang wird durchgeführt, wenn ein vorherbestimmtes Schaltsignal der Einrichtung 90 zugeführt wird. Die Information RA12 bis RA15 zeigt auch wiederholte Reihenfolgen an, wie die vorstehend beschriebenen, jedoch ist die Reihenfolge umgekehrt gegenüber derjenigen, welche durch die Information A12 bis A15 angezeigt ist. Folglich ist es leicht, wie ein Vergleich der Fig. 17E und 17F zeigt, das Punkt muster der Fig. 17F lediglich als ein Ergebnis des Änderns der Anordnungsreihenfolge des Punkt musters der Fig. 17E zu erhalten. Folglich ist es möglich, eine Variation zu den Bildteil-Korrekturvorgängen, wie sie von der Punkt korrektur einheit 7 durchgeführt werden können, durch Hinzufügen einer einfachen Einrichtung, wie der Einrichtung 90 vorzunehmen.

Die vorerwähnte Einrichtung 90 in Fig. 15 kann durch die in Fig. 16 dargestellte Schaltungsformation verwirklicht werden. Die Einrichtung 90 enthält zwei UND-Elemente 91 und 92, ein NICHT-Element 93 und ein ODER-Element 94. Die Einrichtung 90 liefert das Signal A, wenn das Schaltsignal den Wert "1" hat und sie liefert das Signal RA, wenn das Schaltsignal den Wert "0" hat.

Nunmehr wird anhand von Fig. 17A bis 17F eine Ausführungsform einer Zacken-Korrekturoperation gemäß der Erfindung beschrieben. Fig. 17A zeigt ein Bild, das

durch relevante eingegebene Video-(Bild-)Daten dargestellt ist. Beispielsweise werden zwei Teile, welche durch strichlierte Ellipsen eingekreist sind, als Abfragepunkte verwendet, an welchen eine Zacken-Korrekturoperation gemäß der Erfindung durchgeführt wird. Fig. 17B zeigt das Bildteil-Muster, das durch Vergrößern der oberen in Fig. 17A dargestellten Abfrage- bzw. Abtaststelle erhalten worden ist. Die Steuereinheit 3 führt Videodaten, welche den schwarzen Punkt darstellen, welcher als Rechteck mit den gekreuzten Linien in dem Muster der Fig. 17B dargestellt ist, der Punkt korrektur einheit 6 zu. Infolge der Datenwiederholfunktion des FI-FO-Speichers 72 werden eine Linie des ursprünglichen Rechtecks und drei Linien von Kopien aus dem FIFO-Speicher 72 erhalten, wie in Fig. 17D dargestellt ist. Dann gibt wegen des Muster-Erkennungsprozesses, wie er oben beschrieben ist, der Speicherblock die 75 Korrekturinformation ab, welche die vier Rechtecklinien darstellt, wie in Fig. 17E für den relevanten Punkt dargestellt ist. Das Muster, das aus dem in Fig. 17E dargestellten vier Rechtecklinien besteht, wird dann verwendet, um den vorerwähnten durch gekreuzte Linien gekennzeichneten schwarzen Punkt in dem in Fig. 17B dargestellten Abfragebildteil zu ersetzen. Da die vier Rechtecklinien entsprechend korrigiert worden sind, ist der Zacken des in Fig. 17A dargestellten Bildes in feinere in Fig. 17E dargestellten Zacken korrigiert, wobei der Umriß der feineren Zacken die schräge Linie zu sein scheint, wenn das relevante Muster in der Größe merklich verkleinert wird. Somit ist die Zackenkorrektur für den relevanten Punkt durchgeführt worden. Durch viermal schnellere Schreibtakte für die Unterabtastrichtung bei einem Bilddruckprozeß paßt sich die Länge in der Unterabtastrichtung (der vertikalen Richtung in der Figur) des relevanten Punktes, der selbst aus den vier Linien von Kopiepunkten, wie in Fig. 17E dargestellt, besteht, der Länge in der Unterabtastrichtung des ursprünglichen Punktes an, wie in Fig. 17B dargestellt ist. In ähnlicher Weise wird bezüglich der unteren Abtaststelle in in Fig. 17A dargestellten Bild der ursprüngliche Punkt an der Abfragestelle durch das in Fig. 17F dargestellte Muster ersetzt. Das Muster der Fig. 17F verwirklicht die schräge Umrißlinie, deren Gradientenrichtung entgegengesetzt zu dem Muster in Fig. 17E ist. Das Muster in Fig. 17F paßt zu dem relevanten Teil in dem in Fig. 17A dargestellten Bild.

Eine 10 Bit Korrekturinformation wird von dem Speicherblock 75 (insbesondere von dem Musterspeicher 752) in der Ausführungsform der Fig. 11, 12, 13 und 15 zugeführt, wie in den Figuren dargestellt ist. Die jeweiligen 10 Bits in jedem Korrekturinformationssatz können anzeigen, ob, schwarz oder weiß, 10 entsprechende Segmente gedruckt werden, die einen relevanten Punkt darstellen. Der relevante eine Punkt ist beispielsweise einer der ersten bis vierten Linien, die in Fig. 17E und 17F dargestellt sind. Die 10 Segmente werden durch Aufteilen des entsprechenden Punktes erhalten, so daß die Unterteilungslinien sich entlang der Unterabtastrichtung erstrecken. In dem Beispiel der Fig. 17E besteht aus Linie aus vier Segmenten und folglich ist die Bitbreite einer Korrekturinformation 4. Die erste Linie wird beispielsweise durch eine 4 Bit Korrekturinformation (1, 1, 1, 1) verwirklicht und die vierte Linie wird beispielsweise durch eine Korrekturinformation (1, 0, 0, 0) realisiert.

In der vor stehend beschriebenen Punkt korrektur einheit 6 in der erfundungsgemäßen Ausführungsform liefert der Speicherblock 75 eine solche Korrekturinfor-

mation, die eine Bitbreite von beispielsweise 10 hat, in bitparalleler Form. Segmente, wie die vorstehend beschriebenen, entsprechen Zeitabschnitten, die als ein Ergebnis des Unterteilens eines vorherbestimmten Zeitabschnitts erhalten worden sind, welcher für jeden Punkt aufbereitet wird. Der vorherbestimmte Zeitabschnitt ist einer, für welchen der Laser Licht abgibt, um so den relevanten Punkt zu realisieren.

Die in Fig. 4 dargestellte Videodaten-Abgabeeinheit 76 empfängt die bitparallele Korrekturinformation, die von dem Speicherblock 75 zugeführt worden ist, und wandelt sie in eine bit serielle Form um, welche dann dem in Fig. 1 dargestellten Printer/Drucker 4 zugeführt wird. Die zugeführte Korrekturinformation wird verwendet, um die in Fig. 3 dargestellte LD-Einheit 50 zu steuern, die in der Schreibeinheit 26 vorgesehen ist. Folglich gibt die Laserdiode der LD-Einheit 50 Licht ab oder stoppt es entsprechend.

Die vorerwähnte Umsetzung der bitparallelen Korrekturinformation in die serielle Form ist notwendig, wenn das Lichtemissions-Ein/Aus-Steuerverfahren, das für die LD-Einheit 50 verwendet ist, eines ist, bei welchem eine Zweitton-Dateneingabe verwendet wird. Wenn das Ein/Aus-Steuerverfahren eines mit Vielton-Daten ist, ist eine bitparallele in eine bit serielle Umsetzung nicht notwendig. In dem letzten Fall wird die bitparallele Korrekturinformation, die von dem Speicherblock 75 zugeführt ist, unmittelbar als Vielton-Daten verwendet, um das Ein/Aus-Steuern der Laserdiode in der LD-Einheit 50 durchzuführen, so daß die Schreibeinheit 26 das entsprechende Punktbild passend schreibt.

Im Rahmen der Erfindung ist es auch möglich, daß irgendwelche Daten, die entweder von dem Tabellen speicher 751 oder dem Musterspeicher 752 in den vorstehend beschriebenen Ausführungsformen zugeführt worden sind, als bitparallele Information verwendet wird, um das Ein/Aus-Steuern der LD-Einheit ähnlich wie in dem Fall unmittelbar durchzuführen, wenn die bitparallele Korrekturinformation für denselben Zweck verwendet wird.

Ferner zeigt eine solche bitparallele Information (einschließlich einer bitparallelen Korrekturinformation) linienförmige Kenndaten (einer Grenzlinie zwischen einem Schwarzpunkt- und einem Weißpunkt-Bereich, welche durch relevante Videodaten in der Bitmap-Formation dargestellt sind), die für jeden Punkt durch den Mustererkennungsprozeß erhalten worden sind, der, wie vorstehend beschrieben, von der Mustererkennungseinheit 74 durchgeführt worden ist. Folglich kann auch solche bitparallele Information zusätzlich zu der Verwendung bei dem Ein/Aus-Steuern der vorerwähnten Laserdiode als Daten verwendet werden, die von der CPU zu verarbeiten sind, um verschiedene Bildbehandlungsprozesse durchzuführen, wie eine Bildvergrößerung und -verkleinerung. Eine Information der Code information, die von der Mustererkennungseinheit 74 zugeführt worden ist, welche durch den Tabellspeicher 751 und durch den Musterspeicher 752 versorgt wird, kann für den vorerwähnten Zweck verwendet werden.

Anhand von Fig. 18 wird nunmehr eine weitere Ausführungsform der Erfindung beschrieben. Hierbei ist eine Bilddaten-Behandlungseinrichtung 79 in der Punktkorrekturseinheit 7 geschaffen, um einen Bildbehandlungsprozeß durchzuführen, wie eine Bildvergrößerung und -verkleinerung, indem die Einrichtung 79 die in der Punktkorrekturseinheit 7 erzeugte Codeinformation ver-

arbeitet. Somit liefert die Punktkorrekturseinheit 7 die behandelten oder sowohl korrigierten und behandelten Videodaten zusätzlich zu den korrigierten Videodaten. In dieser Blockformation in der Punktkorrekturseinheit 7 können die Zacken-Korrekturoperation und die Bild Behandlungsoperation gleichzeitig parallel an von der Steuereinheit 3 zugeführten Videodaten durchgeführt werden. Folglich können die Ergebnisse der vorstehend angeführten beiden Operationsarten gleichzeitig erhalten werden.

Andere Blockformationen der Punktkorrekturseinheit 7, welche in anderen in Fig. 19 und 20 dargestellten Ausführungsformen der Erfindung mit der Bilddaten Behandlungseinrichtung 79 versehen sind, sind im Rahmen der Erfindung möglich. Somit ist es möglich, einen Zacken-Korrekturprozeß an den behandelten Videodaten durchzuführen, welche von der Einrichtung 79 zugeführt worden sind. Der Unterschied zwischen den in den beiden Figuren dargestellten Formationen ist die Anordnung der Einrichtung 79. In der Ausführung der Fig. 19 ist es möglich, die von der Einrichtung 79 zugeführten Daten für einen anderen Zweck außerhalb der Punktkorrekturseinheit 7 zu verwenden. Eine Mustererkennungs-Verarbeitungseinheit 70 in den zwei Ausführungen enthält einen P/S-Umsetzer 71, einen FIFO-Speicher 72, eine Fenstereinheit 73, eine Mustererkennungseinheit 74 und eine Zeitsteuereinheit 77, wie in Fig. 4 dargestellt ist. Ein Multiplexer MP2 wählt einen der Wege zu der Videodaten-Ausgabeeinheit 76 und der Einrichtung 79 entsprechend den Fällen, bei welchen relevante Ausgabedaten der Bildbehandlungsoperation zu unterziehen sind oder nicht. Ein weiterer Multiplexer MP1 wählt die Ausgangsdaten der Einrichtung 79, wenn der Multiplexer MP2 den Weg zu der Einrichtung 79 wählt.

Nunmehr wird eine andere Blockinformation in einer in Fig. 21 dargestellten Ausführungsform der Erfindung beschrieben. Die Formation enthält eine Fensterbereich-Einstelleinrichtung 80, um eine Anzahl Fensterbereiche in einem Bitmap-Format von Bilddaten zu setzen. Die Formation weist ferner eine Fensterbereich-Bilddatenprozeß-Einstelleinrichtung 81 auf, um einzustellen, ob die Zacken-Korrekturoperation an den jeweiligen mittels der Einrichtung 80 eingestellten Fensterbereichen durchgeführt wird oder nicht. Mit anderen Worten, die Fensterbereich-Einstelleinrichtung 80 setzt über eine CPU u.ä. Bildebenen-Koordinatendaten, die dem jeweiligen Fensterbereich entsprechen. Die Prozeßeinstelleinrichtung 81 benutzt ebenfalls eine CPU u.ä.

Fig. 22 zeigt eine Bildseite, welche durch Ausdehnung von eingegebenen Bilddaten auf das Bitmap-Format erhalten worden ist. Die Bildseite enthält vier Linien, die von dem oberen zu dem unteren Teil der Seite angeordnet sind, wie in Fig. 22 dargestellt ist. Fenster 0, 1, 2 und 3, die durch gestrichelt wiedergegebene Rechtecke angezeigt sind, werden mittels der Fensterbereich-Einstelleinrichtung 80 gesetzt. Die Prozeßeinstelleinrichtung 81 setzt sie so, daß die Fensterbereiche 0 und 2 dem Zacken-Korrekturprozeß unterzogen werden sollten, und die anderen nicht. Fig. 22 zeigt das Ergebnis. Folglich weisen die Linien in den Fensterbereichen 0 und 2 die langen, glatten schrägen Linien auf, während die Linien in den Fensterbereichen 1 und 3 die verschiedenen Stufen (Zacken) enthalten, wie in Fig. 24 dargestellt ist.

Eine Punktkorrekturseinheit 7 wird in einer weiteren Ausführungsform der Erfindung anhand der Fig. 23 und 24 beschrieben. Die Einheit 7 enthält ferner eine Fensterpriorität-Setzeinrichtung 82, um festzusetzen, wel-

cher Prozeß unter den mittels der Einrichtung 81 eingestellten Prozessen für eine Anzahl Fensterbereiche ausgewählt wird. Ein derartiges Auswählen ist notwendig, wenn eine Anzahl Fensterbereiche einander überlappen. Die Einrichtung 82 verwendet eine CPU u.ä. für den vorstehend angeführten Zweck. Die vorstehende Aufgabe der Einrichtung 82 wird anhand von Fig. 24 beschrieben. Fig. 24 zeigt ein Beispiel einer Bildseite, welche durch Dehnen von Bilddaten auf das Bitmap-Format erhalten worden ist. Das relevante Bild enthält den Buchstaben "A". Vier Fensterbereiche 0, 1, 2 und 3 werden mittels der Einrichtung 80 gesetzt, um so Teile des Buchstabens A einzurahmen. Eine Verarbeitung wird in den Fensterbereichen mittels der Prozeßeinstelleinrichtung 81 durchgeführt, so daß die Fensterbereiche 1 und 3 dem Zacken-Korrekturprozeß unterzogen werden sollten, und die anderen nicht. Ferner wird ein Setzen einer Priorität für die Fensterbereiche mittels der die Priorität setzenden Einrichtung 82 durchgeführt, so daß die Priorität folgende ist, 1) Fensterbereich 3, 2) Fensterbereich 2, 3) Fensterbereich 1 und 4) Fensterbereich 0. Fig. 24 zeigt das Ergebnis der Zacken-Korrektur, die gemäß dem Prioritätsetzen durchgeführt worden ist. Die Bereiche, welche schraffiert sind, sind welche, wo die zwei Fensterbereiche einander überdecken.

Ein Punktkorrektureinheit 7 einer weiteren Ausführungsform der Erfindung wird nunmehr anhand von Fig. 25 und 26A bis 26E beschrieben. Die Einheit 7 weist ferner eine Fensterbereich-Bilddaten-Verarbeitungsart-Einstelleinrichtung 23 auf, um spezielle Verarbeitungsarten in Verbindung mit dem Zacken-Korrekturprozeß für jeden Fensterbereich einzustellen. Die Einrichtung 83 verbindet eine CPU u. a. für den vorstehend angeführten Zweck. Die Funktion der Einrichtung 83 wird anhand von Fig. 26 beschrieben. Fig. 26 zeigt ein Beispiel einer Bildseite, die durch Dehnen von Bilddaten in das Bitmap-Format erhalten worden sind. Das relevante Bild enthält vier Buchstaben "A", die von oben nach unten angeordnet sind. Vier Fensterbereiche 0, 1, 2 und 3 werden mittels der Fensterbereich-Einstelleinrichtung 80 gesetzt, um so die jeweiligen vier Buchstaben A einzurahmen. Eine Verarbeitung wird an den Fensterbereichen mittels der eine Verarbeitung einstellenden Einrichtung 81 durchgeführt, so daß die Fensterbereiche 1, 2 und 3 dem Zacken-Korrekturprozeß unterzogen werden sollten und der Fensterbereich 0 nicht. Folglich wird der Buchstaben A in dem Fensterbereich 0 nicht mittels des Zacken-Korrekturprozesses behandelt, wie in Fig. 26B dargestellt ist. Ferner wird die spezielle Verarbeitungsart-Einstellung für die Fensterbereiche, welche der Zacken-Korrektur unterzogen werden, mittels der Verarbeitungsart — Einstelleinrichtung unterzogen. Folglich wird bei dem Fensterbereich 1 eine Methode eingesetzt, bei welcher mittlere Korrekturdaten verwendet werden; bei dem Fensterbereich 2 wird eine Methode eingesetzt, bei welcher dunkle Korrekturdaten verwendet werden, und bei dem Fensterbereich 3 wird eine Methode eingesetzt, bei welcher helle Korrekturdaten verwendet werden. Die vorstehend angeführten Methoden werden nunmehr beschrieben. Bei der Methode, bei welcher mittlere Korrekturdaten verwendet werden, wird eine Korrekturinformation benutzt, so daß weder eine Liniendickung noch eine Liniendünnung durchgeführt wird, und es wird nur eine entsprechende Zacken-Korrektur durchgeführt, wie in Fig. 26C dargestellt ist. Bei der Methode, bei welcher dunkle Korrekturdaten verwendet werden, wird nur eine Korrekturinformation benutzt, so daß ein

Liniendickungsprozeß und auch eine entsprechende Zackenkorrektur durchgeführt wird, wie in Fig. 26D dargestellt ist. Bei der Methode, bei welcher helle Korrekturdaten verwendet werden, wird eine Korrekturinformation benutzt, so daß ein Liniendickungsprozeß und auch eine entsprechende Zacken-Korrektur durchgeführt wird, wie in Fig. 26E dargestellt ist.

Nunmehr wird die Funktion der Zeitsteuereinheit 77 beschrieben, welche ein FGATE-Signal, eine LGATE-Signal, ein LSYNC-Signal, ein Bildtaktsignal WCLK und ein Rücksetzsignal RESET von der Antriebssteuereinheit 4 aus erhalten wird, und welche Taktsignale usw. an den Blöcken 71 bis 76 erzeugt, um eine Synchronisierung vorzunehmen. Das FGATE-Signal legt einen Zeitabschnitt zum Schreiben einer Seite fest. Das LGATE-Signal legt einen Zeitabschnitt zum Schreiben einer Linie fest. Das LSYNC-Signal zeigt Schreibstart- und Endzeitpunkte für jede Linie an. Das Signal WCLK definiert einen Abschnitt zum Lesen und Schreiben eines Punktes. Ferner weist die Zeitsteuereinheit 77 die vorerwähnte Zeitsteuersignal erzeugende Einrichtung auf.

Eine Operationstaktsignal liefert Taktimpulse an die Zeitsteuereinheit 77 und legt den grundsätzlichen Operationsfortschritt der Einheit 77 fest. Das vorerwähnte Takt signal unterscheidet sich von demjenigen, die von der Antriebsansteuereinheit 4 geliefert worden sind und ist entweder ein Steuersignal, das von einer Steuersignal erzeugenden Einrichtung 78 geliefert wird, die in der PunktkorrekturEinheit 7 vorgesehen ist, oder ist ein Steuersignal, das von einer entsprechenden Signalerzeugenden Einrichtung geliefert wird, die außerhalb der PunktkorrekturEinheit 7 vorgesehen ist. Die Steuersignale erzeugende Einrichtung 78 weist, wenn sie vorgesehen ist, einen spannungsgesteuerten Oszillator auf. Die ein entsprechendes Signal erzeugende Einrichtung weist, wenn sie außerhalb der Einheit 7 vorgesehen ist, einen spannungsgesteuerten Oszillator, einen Quarzoszillator o. ä. auf.

Anhand von Fig. 27, 28A bis 28F, 29 und 30A bis 30I wird die ein Zeittaktsignal erzeugende Einrichtung in den Ausführungsformen der Erfindung beschrieben. Die in Fig. 27 dargestellte, ein Zeittaktsignal erzeugende Einrichtung hat einen Aufbau, um die in Fig. 6A bis 6I dargestellte Operation zu erreichen, wie noch beschrieben wird. Die in Fig. 27 dargestellte Schaltungsanordnung weist ein D-Flip-Flop-Element 281, ein Nicht-Element 782 und NAND-Verknüpfungselemente 783 und 784 auf. Diese Schaltung liefert die in Fig. 28A bis 28F dargestellten Signale. Nur in einem Zeitabschnitt, in welchem das FGATE-Signal den Wert "0" hat, d. h. nur in einem Zeitabschnitt, in welchem eine Bildseite zu schreiben/zu drucken ist, ändert sich der Pegel des DATA-SEL-Signals abwechselnd zwischen dem hohen und dem niedrigen Pegel jeweils an den Anstiegsflanken des Signals LSYNC. Während eines Zeitabschnitt während welchem das FGATE-Signal den Wert "0" hat, während eines Zeitabschnitts außerhalb des Zeitabschnitts, in welchem eine Bildseite geschrieben wird, bleibt das DATA-SEL-Signal auf dem Wert "1". Ferner liefern die Verknüpfungselemente 783 und 784 ein LSYNC-OUT-Signal bzw. eine LGATE-OUT-Signal. Die Verknüpfungselemente 783 und 784 geben das LSYNC- und das LGATE-Signal ab, die beide von der Antriebsansteuereinheit 4 zugeführt werden. Dieses Steuern benutzt das DATA-SEL-Signal, wie in Fig. 27 dargestellt ist. Die auf diese Weise erzeugten LSYNC-OUT- und LGATE-OUT-Signale mit den in Fig. 28E und 28F dargestellten Wellenformen werden der Steuereinheit 3 zugeführt.

Beispielsweise sollen die LSYNC- und die LGATE-Signale einen sein, welche eine Auflösung von 300 dpi in der Hauptabtastrichtung bzw. 600 dpi in der Unterabtastrichtung in einem endgültigen Bild ergeben. Dieses endgültige Bild wird durch Ausdrucken über einen Laserdrucker 2 mit Hilfe der LSYNC- UND LGATE-Signale erhalten, welche von der Steuereinheit 4 zugeführt worden sind. Die LSYNC-OUT- und LGATE-OUT-Signale werden verwendet, um eine Zacken-Korrekturoperation freizugeben, wie sie vorstehend anhand der Fig. 17A bis 17F beschrieben worden ist, was unter der Voraussetzung durchgeführt wird, daß die grundsätzlichen Auflösungs-Kenndaten von 300 dpi in der Hauptabtastrichtung bzw. von 600 dpi in der Unterabtastrichtung in Abhängigkeit von der Möglichkeit des Laserdruckers 2 nicht erhöht werden müssen. Wie aus Fig. 28B und 28D bis 28Z zu erkennen ist, erscheint ein abklingender Impuls entweder in dem LSYNC-OUT- oder in dem LGATE-OUT-Signal, während in dem Zeitabschnitt zwei aufeinanderfolgende abklingende Impulse entweder in dem LSYNC- oder dem LGATE-Signal erscheinen. D.h. die Impulshäufigkeit entweder des LSYNC-OUT- oder des LGATE-OUT-Signals ist die Hälfte der Häufigkeit des LSYNC- oder des LGATE-Signals. Folglich sind die LSYNC-OUT- und die LGATE-OUT-Signale einen, damit die Steuereinheit 3 Videodaten verarbeitet/erzeugt, welche Datenkennwerte haben, welche der Auflösung von 300 dpi in der Hauptabtastrichtung bzw. von 300 dpi in der Unterabtastrichtung unter der vorstehend wiedergegebenen Annahme entsprechen. D.h. die Auflösung ist die Hälfte der oben angeführten Auflösung in der Unterabtastrichtung.

Die Steuereinheit 3 liefert folglich Videodaten mit den vorstehenden Datenkennwerten. Ein derartiger Videodatenwert ist einer, der zu dem endgültig gedruckten Bild führt, das aus Linien der halben Dichte besteht, wenn derartige Videodaten so, wie sie sind, verwendet werden, um das Bild in dem Laserdrucker 2 zu erzeugen. Die halbe Dichte ist eine Dichte, welche die Hälfte der Dichte ist, welche den Auflösungs-Kennwerten von 300 dpi in der Hauptabtastrichtung bzw. von 600 dpi in der Unterabtastrichtung entspricht. In diesem Fall verarbeitet die Steuereinheit 3 Videodaten mit den Kennwerten, welche der halben Auflösung von 300 dpi in der Hauptabtastrichtung bzw. von 600 dpi in der Unterabtastrichtung entsprechen.

Die Steuereinheit 3 liefert Videodaten nur während Zeitabschnitten, während welcher das in Fig. 28F dargestellte LGATE-OUT-Signale den Wert "0" an den in Fig. 4 dargestellten FIFO-Speicher 72 sendet. In den in Fig. 28F dargestellten Abschnitten sind Ziffern 1, 2 und 3 eingetragen. Diese zeitliche Steuerung der Videodaten-Abgabe entspricht derjenigen, welche in Fig. 6A dargestellt ist. Die Ziffern 1, 2 und 3 entsprechen denselben in Fig. 6A dargestellten Ziffern. In Fig. 28A bis 28F ist der Zeitabschnitt, in welchen das FGATE-Signal den Wert "0" hat, verkürzt, d. h. die einzelnen Abschnitte, in welcher das FGATE-Signal den Wert "0" hat, enthält nur sechs Abschnitte, in welchen das LSYNC-Signal den Wert "0" hat, wobei dies in den Figuren nur der Einfachheit halber so dargestellt ist. Dann bewirkt der FIFO-Speicher 72 eine Operation wie diejenige, welche in Fig. 6A bis 6I dargestellt ist, so daß, wie oben beschrieben, Sätze von Videodaten, die von der Steuereinheit 3 angelegt worden sind, wiederholt werden. Folglich werden zwei identische Zeilen nacheinander in die Zeilenspuffer in dem FIFO-Speicher 72 geschrieben, und die Videodaten werden folglich mit Hilfe der wiederholten

Daten Zeile für Zeile vervollständigt. Folglich erhält die Punktkorrekturseinheit 7 Videodaten, welche der Auflösung von 300 dpi in der Hauptabtastrichtung/300 dpi in der Unterabtastrichtung entsprechen und liefert folglich Videodaten, welche der Auflösung von 300 dpi in der Hauptabtastrichtung bzw. von 600 dpi in der Unterabtastrichtung entsprechen.

In der in Fig. 27 dargestellten Ausführungsform kann das in Fig. 27 und 28C dargestellte DATA-SEL-Signal als eine wiederholte Codeinformation, wie A12 bis A15 verwendet werden, die in Fig. 11 dargestellt ist, usw. Wie in Fig. 6B und 6C dargestellt, hat das DATA-SEL-Signal den Wert "1", wenn ein ursprünglicher Satz von Videodaten angelegt wird, und das DATA-SEL-Signal hat den Wert "0", wenn die Kopie des ursprünglichen Satzes von Videodaten angelegt wird.

Anhand von Fig. 29 und 30 wird nunmehr die ein Zeitsteuersignal erzeugende Einrichtung in einer weiteren Ausführungsform gemäß der Erfindung beschrieben. Die in Fig. 29 dargestellte ein Zeitsteuersignal erzeugende Einrichtung hat einen Aufbau, um die in Fig. 7A bis 7I dargestellte Operation zustandezubringen. In Fig. 30A bis 30I ist der Zeitabschnitt, in welchem das FGATE-Signal den Wert "0" hat, verkürzt, d. h. der einzige Abschnitt, in welchem das FGATE-Signal den Wert "0" hat, weist, um die Darstellung in den Figuren zu vereinfachen, nur acht Perioden auf, innerhalb welcher das LSYNC-Signal den Wert "0" hat. Die das Zeitsteuersignal erzeugende Einrichtung enthält 4 Bit Zähler 785, 786 und D-Flip-Flops 787, 788 und arbeitet so, wie in Fig. 30A bis 30I dargestellt ist. Bei dieser Arbeitsweise zählen die 4 Bit Zähler jede Anstiegsflanke in dem LSYNC-Signal nur in einem Zeitabschnitt, in welchem das FGATE-Signal den Wert "0" hat, d. h. in einem Zeitabschnitt, in welchem die relevanten Bilddaten verwendet werden, um die entsprechende Bildseite zu schreiben. Der Zähler 785, welcher COUNT-A-, COUNT-B-, COUNT-C-UND COUNT-D-Signale als zu ladende Daten empfängt, gibt folglich die vorstehend beschriebenen Daten A12 bis A18 als Daten ab, die sich aus dem Inkrementieren der vorstehenden, geladenen Daten ergeben, wie in Fig. 29 dargestellt ist.

In dem in Fig. 30A bis 30I dargestellten Beispiel zeigen die COUNT-A- bis COUNT-D-Daten den Wert "C(12)", und folglich der Zähler 785 nacheinander entsprechend seiner Zählfunktion, wie in Fig. 30E dargestellt ist, die Daten A12 bis A15 ab, die "C(12)", "D(13)", "E(14)" und "F(15)" anzeigen. Parallel dazu geben der Zähler 786 und die mit dem Zähler 786 verbundenen Inverter, die in Fig. 29 dargestellt sind, nacheinander entsprechend ihrer Zählfunktion, wie in Fig. 30F dargestellt ist, die vorerwähnten Daten RA12 bis RA15 ab, welche die Werte "F(15)", "E(14)", "D(13)" und dann "C(12)" anzeigen. Folglich geben die Zähler 785 und 786 Daten ab, welche umgekehrte Reihenfolgen anzeigen. Das in Fig. 29 und 30I dargestellte Signal DATA-SEL hat den Wert "0" nur dann, wenn die von dem Zähler 785 abgegebenen Daten A12 bis A18 eins sind, was denselben Wert anzeigt, welcher durch die Ladeden COUNT-A bis COUNT-D angezeigt werden ist.

Die 4 Bit Zähler 785 bis 786 zählen nicht, wenn das LGATE-Signal den Wert "1" hat, d. h. wenn es außerhalb der Zeitabschnitt liegt, werden die relevanten Bilddaten verwendet, um das entsprechende Seitenbild zu schreiben. Der 4 Bit Zähler 785 lädt die Signale COUNT-A bis COUNT-D, wenn das FGATE-Signal den Wert "1" hat. Folglich werden die Ausgangsdaten A12 bis A15 auf dem Wert gehalten, welcher derselbe ist, wie derjenige,

welcher durch die Ladedaten COUNT-A bis COUNT-D angezeigt ist, d. h. "C (12)" in diesem Zeitabschnitt. Die Ausgangsdaten RA12 bis RA15 werden in diesem Zeitabschnitt auf dem Wert "F (15)" gehalten. Der Grund hierfür wird nunmehr beschrieben. Wie in Fig. 29 dargestellt, sind alle Ladeanschlüsse D0 bis D3 des Zählers 786 geerdet. Folglich hat das Datenladen den Wert "0". Dementsprechend wird in dem Zähler 785 das Datenladen von dem Zähler 786 durchgeführt, wenn das FGA-TE-Signal den Wert "1" hat. Folglich sind alle Signale, welche über die Anschlüsse Q0 bis Q3 abgegeben werden, "0". Das heißt  $(Q_0, Q_1, Q_2, Q_3) = (0, 0, 0, 0)$ . Die Signale werden dann durch die Inverter invertiert, welche mit den Anschlüssen Q0 bis Q3 verbunden sind, wie in Fig. 29 dargestellt ist. Folglich hat jedes der sich ergebenden Signale RA12 bis RA15 den Wert "1". D.h.  $(RA_{12}, RA_{13}, RA_{14}, RA_{15}) = (1, 1, 1, 1)$ , was in der hexadezimalen Schreibweise "f (15)" entspricht. Das Signal DATA-SEL wird in diesem Zeitabschnitt auf dem Wert "0" gehalten, da die vorerwähnte relevante Bedingung in der in Fig. 29 dargestellten Schaltungsanordnung voll erfüllt ist.

Ein LSYNC-OUT- und ein LGATE-OUT-Signal, die in Fig. 30G und 30H dargestellt sind, werden in der in Fig. 29 dargestellten Zeitsignal erzeugenden Einrichtung erzeugt, wobei Verknüpfungsglieder für die LSYNC- und LGATE-Signale verwendet werden, welche von der Ansteuereinheit 4 zugeführt worden sind.

In einem Beispiel, das dem vorstehend beschriebenen Beispiel ähnlich ist, ist angenommen, daß die LSYNC- und die LGATE-Signale einsen sind, was bei einem endgültigen Bild zu einer Auflösung von 300 dpi in der Hauptabtastrichtung / von 600 dpi in der Unterabtastrichtung führt. Das endgültige Bild wird durch Ausdrucken über den Laserdrucker 2 erhalten, wobei die von der Ansteuereinheit 4 zugeführten LSYNC- und LGATE-Signale verwendet werden. Die LSYNC-OUT- und LGATE-OUT-Signale werden verwendet, um eine Zacken-Korrekturoperation zu ermöglichen, wie sie vorstehend anhand der Fig. 17A bis 17F beschrieben ist, was unter der Bedingung entsprechend durchgeführt wird, daß die grundlegenden Auflösungskennwerte bei einer Auflösung von 300 dpi in der Hauptabtastrichtung bzw. Von 600 dpi in der Unterabtastrichtung in Abhängigkeit von der Möglichkeit des Laserdruckers 2 nicht erhöht zu werden brauchen. Wie aus Fig. 30B, 30D und 30H zu ersehen ist, erscheint ein abklingender Impuls entweder in dem LSYNC-OUT- oder dem LGATE-OUT-Signal in dem Zeitabschnitt, in welchem vier aufeinanderfolgende abklingende Impulse entweder in dem LSYNC- oder dem LGATE-Signal erscheinen. D.h. die Impulshäufigkeit entweder des LSYNC-OUT- oder des LGATE-OUT-Signals ist ein Viertel der Häufigkeit des LSYNC- oder LGATE-Signals. Folglich sind die LSYNC-OUT- und die LGATE-OUT-Signale einsen, wodurch die Steuereinheit 3 Videodaten verarbeitet/erzeugt, welche Datenkennwerte haben, welche einer Auflösung von 300 dpi in der Hauptabtastrichtung bzw. von 150 dpi in der Unterabtastrichtung unter der vorstehenden Annahme entsprechen. D.h. die Auflösung ist ein Viertel der vorstehend angegebenen in der Unterabtastrichtung.

Die Steuereinheit 3 liefert folglich Videodaten mit den vorstehenden Datenkennwerten. Solche Videodaten sind eins, was zu dem endgültig gedruckten Bild führt, das aus Zeilen mit der 1/4-Dichte besteht, wenn solche Videodaten, so wie sie sind, verwendet werden, um das Bild in dem Laserdrucker 2 zu erzeugen. Die

1/4-Dichte ist ein Viertel der Dichte, die den Auflösungskennwerte von 300 dpi in der Hauptabtastrichtung bzw. von 600 dpi in der Unterabtastrichtung entspricht. In diesem Fall verarbeitet die Steuereinheit 3 Videodaten mit den Kennwerten, die einem Viertel der Auflösung von 300 dpi in der Hauptabtastrichtung bzw. von 600 dpi in der Unterabtastrichtung entsprechen.

Die Steuereinheit 3 liefert Videodaten nur während Zeitabschnitten, in welchen das in Fig. 30H dargestellte LGATE-OUT-Signal den Wert "0" hat, an den in Fig. 4 dargestellte FIFO-Speicher 72. In den in Fig. 28f dargestellten Abschnitten sind Ziffern 1 und 2 eingetragen. Diese zeitliche Steuerung der Videodaten-Zufuhr entspricht der in Fig. 7A dargestellten. Die vorstehenden Ziffern 1 und 2 entsprechen denselben in Fig. 7A dargestellten Ziffern. Der FIFO-Speicher 72 bewirkt die Operation, wie sie in Fig. 7A bis 7E dargestellt ist, so daß, wie oben beschrieben, Videodatensätze, die von der Steuereinheit 3 zugeführt worden sind, dreimal wiederholt werden. Folglich empfängt die Datenkorrekturseinheit 7 Videodaten, welche der Auflösung von 300 dpi in der Hauptabtastrichtung bzw. von 150 dpi in der Unterabtastrichtung entsprechen und liefert folglich Videodaten, welche der Auflösung von 300 dpi in der Hauptabtastrichtung von 600 dpi in der Unterabtastrichtung entsprechen.

In der Fig. 29 dargestellten Ausführungsform können die in Fig. 29 und 30E dargestellten Daten A12 bis A15 als in einer wiederholten Reihenfolge wiedergegebene Codeinformation verwendet werden, wie beispielweise die in Fig. 11 dargestellten Daten A12 bis A15, usw. die vorstehenden Werte "C", "D", "E" und "F" der Daten A12 bis A15 zeigen die wiederholten Reihenfolgen der relevanten Videodaten an, d. h. die ersten, zweiten, dritten und vierten zugeführten Videodaten.

Die Korrekturdaten, die vorher in dem in Fig. 11 dargestellten Musterspeicher 752 gespeichert sind, können vorher selektiv entweder durch die MPU 31 in der in Fig. 1 dargestellten Steuereinheit 3 oder durch die CPU 41 in der Ansteuereinheit 4 mit Hilfe von in dem RAM 32 oder 42 gespeicherten Daten geladen werden. Andererseits kann der Host-Computer 1 auch die notwendigen Daten in den Musterspeicher 752 laden. Folglich können verschiedene Korrekturdaten einfach durch Ändern von gerade geladenen Daten beliebig verwendet werden.

Eine Technologie, welche in den vorstehend beschriebenen Ausführungsformen der Erfindung verwendet ist, wird nunmehr beschrieben; diese Technologie ist auch in den Eingangs erwähnten, offengelegten japanischen Patentanmeldung Nr. 5-207 282 verwendet. Anhand von Fig. 8 und 31A bis 41 ist eine Fensterbereich-Trennoperation vor einer Musteranpassungsoperation durchzuführen; Muster sind zu fühlen und Bereiche sind für das Musterfühlen zu verwenden.

Fenster 73, wie sie vorstehend erwähnt sind, werden nunmehr beschrieben. Jeder Fensterbereich ist durch ein Abfragefenster mit Abmessungen von 7 (Höhe) x 11 (Länge) festgelegt, wie in Fig. 8 dargestellt ist und wird durch die in Fig. 5 dargestellten vorstehend beschriebenen durch die 7 Zeilen-Schieberegister 73A bis 73g verwirklicht. Jede Zeile eines Schieberegisters enthält 11 Bit Register, wie in Fig. 5 dargestellt ist. Folglich liefern die 7 Zeilen-Schieberegister 73a bis 73g, 77 Bit Daten  $(7 \times 11)$  entsprechend den in Fig. 8 dargestellten 77 Punkten  $(7 \times 11)$ .

Unter den 77 Punkten werden 49 Punkte, die durch eine in Fig. 8 dargestellte gestrichelte Linie eingerahmt

sind, verwendet, um spezifische Muster zu fühlen, d. h. es sind annähernd vertikale oder annähernd horizontale Linien. Solche zu fühlenden Linien sind Grenzlinien zwischen einem Schwarzpunkt- und einem Weißpunkt-Bereich.

Nunmehr werden Kernbereiche beschrieben. Der Kernbereich im Beispiel der Fig. 8 ist der Kernbereich 73c mit den Abmessungen von  $(3 \times 3)$  Punkten, welche in Fig. 8 durch eine ausgezogene Linie umrahmt. Der zentrale Punkt in den  $(3 \times 3)$  Punkten ist ein relevanter Punkt, der gerade bei der Zacken-Korrekturoperation verarbeitet wird.

Fig. 31A bis 33G zeigen Linienmuster mit Gradienten (Steigungen) von  $45^\circ$  (1/1). Solche Linienmuster sind nicht solche, die bei der Zacken-Korrekturoperation in den vorstehend beschriebenen Ausführungsformen zu korrigieren sind. Zu extrahierende Linienmuster, die bei der Zackenkorrektur-Operation zu korrigieren sind, sind entweder annähernd horizontal mit Gradienten, die gleich oder kleiner als die Neigung von  $1/2$  sind, oder sind annähernd horizontal mit Gradienten, die gleich oder mehr als die Neigung von  $2/1$  sind. In den Ausführungsformen sind solche Arten Linienmustern angenommen, die in den endgültigen Bildern von Benutzern als Zacken (jags) zu erkennen sind. In den Ausführungsformen werden annähernd horizontale und annähernd vertikale Linien erkannt, wie beispielsweise solche, die vorstehend erwähnt sind. Ein Unterschied zwischen den zwei Erkennungsprozessen liegt lediglich darin, daß ein Linienmuster dem anderen Linienmuster entspricht, wenn eines um einen Winkel von  $90^\circ$  gedreht wird. Folglich werden nachstehend der Einfachheit halber hauptsächlich nur Fälle für annähernd horizontale Linien beschrieben.

Fig. 32A bis 32G veranschaulichen annähernd horizontale Linienmuster. Linienmuster, die Gradienten haben, die klein oder kleiner als die Neigungen von  $1/2$  sind, sind die folgenden zwei Musterarten in solchen  $(3 \times 3)$  Kernbereichen. Erste Musterarten sind solche, die in Fig. 33B, 32C, 32E und 32F dargestellt sind.

Die in diesen Figuren dargestellten Punktmustern entsprechen den Gradientenlinien mit bestimmten Gradientengraden, welche durch die relevanten Bilddaten dargestellt sind. Die vorerwähnten Grade sind mehr als der Grenzwert zum Darstellen der Bilddaten mit den rechts horizontal angeordneten Punktreihen. Folglich ist jede der sich ergebenden Punktreihen eine, in welcher schwarze Punkt in zwei verschiedenen Schritten/Höhen angeordnet sind. Die zweiten Musterarten sind solche, die in Fig. 32A, 32C und 32G dargestellt sind, wobei jedes dieser Muster eines ist, in welchem schwarze Punkte in einer einzigen Stufe/Höhe angeordnet sind. Fig. 33A bis 33C stellen zwei ähnliche Musterarten für annähernde vertikale Linien dar.

Bei den Ausführungsformen bei der Erfindung werden die in Fig. 31A bis 33G dargestellten Muster vorher gespeichert, die verwendet werden, um zu prüfen, ob diese gespeicherten Muster zu einem Muster passen, das in einem Kernbereich 73C von tatsächlich eingegebener Bilddaten erscheint. Folglich kann das relevante Muster der eingegebenen Bilddaten ohne weiteres in einen klassifiziert werden, welches in den Ausführungsformen nicht korrigiert werden sollte, in eines, welches ein Kandidat einer annähernd horizontalen Linie ist, oder in eines, welches ein Kandidat einer annähernd vertikalen Linie ist.

Nunmehr werden periphere Bereiche beschrieben, welche, wie vorstehend erwähnt, verwendet werden, um

festzuzeigen, daß Kandidaten von annähernd horizontalen oder vertikalen Linien, wie vorstehend erwähnt ist, tatsächliche annähernd horizontale oder vertikale Linien sind, welche in den Ausführungsformen korrigiert werden sollen. Wie vorstehend bezüglich der Fig. 34A bis 34D beschrieben, werden die peripheren Bereiche 73R, 73L, 73U und 73D, die rechts, links, oben und unten von dem Kernbereich 30c angeordnet sind, verwendet. Wie in den Figuren dargestellt, hat jeder dieser vier peripheren Bereiche 73R, 73L, 73U und 73D zwei Punkte, die an deren Enden angeordnet sind, an welchen Punkten die aneinander grenzenden zwei Bereiche übereinander überdecken.

Wie in Fig. 35A bis 35C und 36A bis 36C dargestellt ist, ist jeder der vier peripheren Bereiche in drei Unterbereiche unterteilt. Jedoch überdeckt jeder dieser Unterbereiche die benachbarten Unterbereiche, wie in den Figuren dargestellt ist. Folglich sind der rechte Bereich 73R und der linke Bereich 73L in rechte Unterbereiche 73Ra, 73Rb und 73Rc und linke Unterbereiche 73La, 73Lb und 73Lc aufgeteilt. Der obere Bereich 73U und der untere Bereich 73D sind in die oberen Unterbereiche 73Ua, 73Ub und 73Uc und in die unteren Unterbereiche 73Da, 73Db und 73Dc aufgeteilt. Eine solche Unterteilung der peripheren Bereiche wird durchgeführt, um die Schaltungsformation zu vereinfachen, die erforderlich ist, um die Musteranpassungsoperation mit Hilfe der auf diese Weise unterteilten peripheren Bereiche durchzuführen. Die Ausführungen wählen entsprechende Unterbereiche unter den vorerwähnten Unterbereichen in Abhängigkeit von dem Zeilenmuster aus, das zwischen einem Schwarzpunkt- und einem Weißpunkt-Bereich vorhanden ist, die in dem Kernbereich 73C gefühlt werden, an welchen die Unterbereiche angrenzen.

Das heißt, wenn das gefühlte Linienmuster in dem Kernbereich 73C der Linie entspricht, die den Gradienten hat, der gleich oder kleiner als die Neigung von  $1/2$  ist, d. h. annähernd horizontal ist; der rechte oder linke Bereich 73L oder 73R, der in Fig. 34A und 34B dargestellt ist, oder die beiden Bereiche werden dann geprüft. In ähnlicher Weise werden dann, wenn die gefühlte Linie in dem Kernbereich 73C der Linie entspricht, welche den Gradienten hat, welcher gleich oder mehr als die Neigung  $2/1$  ist, d. h. annähernd vertikal ist, die oberen oder unteren Bereiche 73U oder 73D, welche in Fig. 34C und Fig. 34D dargestellt sind, oder die zwei Bereiche überprüft. Ferner wird bei einer solchen Überprüfung in Abhängigkeit von der Position der gefühlten Linie in dem Kernbereich 73C ein entsprechender Unterbereich in dem vorerwähnten Bereich ausgewählt, um die Effizienz bei der Operation zu überprüfen, bei welcher festgestellt wird, ob die relevante Linie einer Linie entspricht oder nicht, die in der Ausführungsform der Erfindung zu korrigieren ist. Bei einem in Fig. 37 dargestellten Beispiel werden der linke Unterbereich 73Lb und der rechte Unterbereich 73Ra geprüft, und bei einem in Fig. 38 dargestellten Beispiel werden der obere Unterbereich 73Ub und der untere Unterbereich 73Dc geprüft. Andererseits ist auch eine andere Unterbereich-Auswahl möglich, bei welcher der rechte Unterbereich 73Ra bei dem Beispiel in Fig. 27 und der obere Unterbereich 73Ub bei dem Beispiel in Fig. 38 geprüft werden.

Der Grund für diese Vorgehensweise wird nunmehr beschrieben. Diese Vorgehensweise basiert auf der Annahme, daß jede Linie, welche durch die seriellen schwarz ausgelegten Kreise in den Fig. 37 oder 38 dargestellt ist, stellt eine gerade Linie mit einem konstanten

Gradienten dar. Im Beispiel der Fig. 37 ist in dem Kernbereich 73C der Höhenunterschied der Zeile auf der linken Seite in dem Bereich 73C festgelegt, wie in Fig. 37 dargestellt ist. Basierend auf der vorstehenden Annahme sollten die beidseitigen anschließenden Höhendifferenzen auf derselben Linie an den Positionen angeordnet sein, die denselben Abstand von der vorerwähnten Höhendifferenz haben, welche in dem Kernbereich 73C vorhanden ist. Folglich ist die rechseitige Höhendifferenz nahe bei den Kanten des Kernbereichs 73C.

Anhand von Fig. 9 werden nunmehr die jeweiligen Ausgangssignale der Blöcke beschrieben, die in der Mustererkennungseinheit 74 vorhanden sind. Signale, die von der den Kernbereich erkennenden Einheit 741 geliefert werden, werden nunmehr beschrieben.

Ein H/V-Signal zeigt an, ob das relevante Muster einer annähernd horizontalen oder vertikalen Linie entspricht oder nicht. Der Pegel des Signals ist die hohe "1", wenn die Linie annähernd horizontal ist, und ist eine niedrige "0", wenn die Linie annähernd vertikal ist.

Signale DIR0 und DIR1 zeigen Zweibit-Codes, wie folgt an:

Wenn beide Signale DIR1 und DIR0 auf "0" sind, zeigt das Signal einen nicht passenden Zustand an (d. h. die relevante Linie sollte nicht korrigiert werden); wenn das Signal DIR1 auf "0" ist und das Signal DIR0 auf "1" ist, zeigt das Signal an, daß die relevante Linien einen Gradienten hat, so daß die rechte Seite höher und die linke Seite niedriger festgelegt ist; wenn das Signal DIR1 auf "1" ist und das Signal DIR0 auf "0" ist, zeigt das Signal an, daß die relevante Linie einen Gradienten hat, so daß die linke Seite höher und die rechte Seite niedriger angeordnet ist, und wenn die zwei Signale auf "1" sind, zeigt das Signal an, daß die relevante Linie eine horizontale oder vertikale ist.

Ein B/W-Signal zeigt an, ob der relevante Punkt ein schwarzer oder ein weißer Punkt ist, d. h. den Inhalt des relevanten Punktes selbst. Wenn der relevante Punkt ein schwarzer ist, ist das Signal auf "1", und es ist auf "0", wenn der relevante Punkt der weiße ist.

Ein U/L-Signal zeigt an, daß, wenn der relevante Punkt ein weißer ist, der relevante Punkt (rechts) höher oder (links) niedriger als die Linie ist (eine Grenzlinie zwischen schwarzen und weißen Bereichen). Das Signal ist auf "1" wenn der Punkt (weiter rechts) höher ist, und das Signal ist auf "0", wenn der Punkt (links) tiefer angeordnet ist.

Ein GST-Signal zeigt an, ob der relevante Punkt einer ist, welcher dem Ausgangspunkt der Linie entspricht, von welchem Punkt aus der Gradient der Linie berechnet wird, d. h. welcher dem Zacken entspricht, der auf der Linie vorhanden ist, d. h. der Position entspricht, an welcher die Linie schrittweise ansteigt/abfällt. Das Signal ist auf "1", wenn der Punkt der Ausgangspunkt ist.

Ein RUC-Signal zeigt an, ob der rechte Bereich 73R oder der obere Bereich 73U geprüft werden sollte. Wenn er geprüft werden sollte, ist das Signal auf "1". Ein LLC-Signal zeigt an, ob der linke Bereich 73L oder der untere Bereich 73 D geprüft werden sollte. Wenn er geprüft werden sollte, ist das Signal auf "1".

Der Zustand, wenn das RUC-Signal auf "1" ist, und auch das LLC-Signal auf "1" ist, zeigen an, daß die relevante Linie die horizontale oder vertikale ist. Der Zustand, bei welchem das RUC-Signal auf "0" und auch das LLC-Signal auf "0" ist, zeigen an, daß eine Musteranpassungsoperation an der relevanten Linie nicht durchgeführt werden muß.

CC0- und CC1-Signale sind eine Zweibit-Information, welche die Zahl, eine von 1 bis 3, von seriellen Punkten anzeigt, die dem Linienmuster entsprechen, das in dem Kernbereich 73C vorhanden ist. RUAS0- und RUAS1-Signale sind eine Zweibit-Information, um einen der drei Unterbereiche in dem rechten Bereich 73R oder dem oberen Bereich 73U zu spezifizieren. LLAS0- und LLAS1-Signale sind eine Zweibit-Information, die eine von den drei Unterbereichen in dem linken Bereich 73L oder dem unteren Bereich 73D spezifiziert.

Signale, die von der einen peripheren Bereich erkennenden Einheit 772 abgegeben werden, werden nunmehr beschrieben. "cn0 bis cn2"-Signale sind eine 3 Bit-Information, welche die Zahl, eine von 0 bis 4, von horizontal oder vertikal verlaufenden seriellen Punkten anzeigt, die in den peripheren Bereichen vorhanden sind, wobei die Punkte anschließend an die spezifischen Punkte folgen, welche in dem Kernbereich 73C vorhanden sind.

"dir0 und dir1"-Signale sind eine 2 Bit-Information, welche die Gradientenrichtung des Linienmusters anzeigt, das als ein Ergebnis der Musteranpassungs-Fühloperation festgestellt worden ist, die an den Unterbereichen durchgeführt worden ist. Die Codierweise entspricht derjenigen bei den vorstehend beschriebenen DIR0- und DIR1-Signalen.

Signale, die von den Multiplexern (MUX) 743 und 744 abgegeben werden sind, werden nunmehr beschrieben.

RUCN0- bis RUCN2-Signale sind eine 3 Bit-Information, welche die Anzahl von seriellen Punkten anzeigt, die in dem rechten Bereich 73R oder dem oberen Bereich 73U vorhanden sind. RUDIR0- und RUDIR1-Signale sind eine 3 Bit-Codeinformation, welche die Gradientenrichtung der Linie anzeigt, die in dem rechten Bereich 73R oder dem oberen Bereich 73U vorhanden ist. LLCN0- bis LLCN2-Signale sind eine 3 Bit-Information, welche die Anzahl an seriellen Daten anzeigt, die in dem linken Bereich 73L oder dem unteren Bereich 73D vorhanden sind. LLDIR0- und LLDIR1-Signale sind eine 3 Bit-Codeinformation, welche die Gradientenrichtung der Linie anzeigt, die in dem linken Bereich 73L oder dem unteren Bereich 73D vorhanden ist.

Signale, die von der Bestimmungseinheit 747 abgegeben werden sind, werden nunmehr beschrieben.

DIR0- und DIR1-Signale sind dieselben wie diejenigen, welche von der den Kernbereich erkennenden Einheit 741 abgegeben werden sind.

Ein Nicht-Anpassungs-Signal zeigt an, wenn es auf "1" ist, daß kein Muster, welches in den Ausführungsformen korrigiert werden sollte, in der gerade erkannten Linie vorhanden ist.

Signale, welche von der Gradienten-Berechnungseinheit 745 zugeführt worden sind, werden nunmehr beschrieben.

G0- bis G3-Signale sind eine 4 Bit-Information, welche den Gradienten der gerade erkannten Linie anzeigt. Dieser Gradient ist jedoch nicht der mathematische Gradient, sondern ist durch die Anzahl der horizontal seriellen Punkte oder die Anzahl der vertikal seriellen Punkte ausgedrückt. D.h. die Anzahl an seriellen Punkten, die vor der Linie vorhanden sind, die einen Punkt schrittweise ansteigt/abfällt, entspricht dem vorerwähnten Gradienten.

Signale, die von der Positions-Berechnungseinheit 726 abgegeben werden sind, werden nunmehr beschrieben.

"p0- bis p3"-Signale sind eine Vierbit-Codeinformation, welche die Position des relevanten Punktes anzeigt.

Die Signale zeigen die Anzahl an Punkten an, welche an dem linksseitigen Ende beginnen und an dem relevanten Punkt innerhalb der seriellen Punkte enden, wenn die Linie eine annähernde horizontale Linie ist, und welche die Anzahl Punkte anzeigen, die an dem unteren Endpunkt beginnen und an dem relevanten Punkt in den seriellen Punkten enden, wenn die Linie eine annähernd vertikale Linie ist.

Signale, die von dem Verknüpfungsglied 748 abgegeben werden sind, werden nunmehr beschrieben.

P0- bis P3-Signale zeigen eine Positions-Codeinformation an und sind die vorstehend wiedergegebenen Signale "p0 bis p3", welche von der Positions-Berechnungseinheit 746 zugeführt werden, wenn das Nicht-Anpassungs-Signal, das von der Bestimmungseinheit 747 zugeführt worden ist, auf "0" ist. Wenn das Nicht-Anpassungs-Signal auf "1" ist, zeigen die P0- bis P3-Signale "0" an.

Operationen, welche von den jeweiligen Blöcken von der in Fig. 9 dargestellten Mustererkennungseinheit 74 durchgeführt sind, lassen sich einfach beschreiben. Die einen Kernbereich erkennende Einheit 741 extrahiert die Daten der Punkte, welche in dem Kernbereich 73C des Fensters 73 enthalten sind, und führt verschiedene Überprüfungs- und Berechnungsoperationen für den relevanten Punkt durch. Folglich gibt die Einheit 741 die vorstehend beschriebenen H/V-, B/W- und U/L-Signale an den Speicherblock 75 ab und ändert den Eingang der Multiplexer 743 und 744 mit Hilfe des Signals H/V, das anzeigt, ob die Linie annähernd horizontal oder annähernd vertikal ist.

Die Einheit 741 legt die RUC- und LLC-Signale, welche anzeigen, welcher peripherer Bereich geprüft werden sollte, an die Recheneinheit 745 und die Bestimmungseinheit 747 an. Die Einheit 741 liefert ferner die GST-Signale, die anzeigen, ob der relevante Punkt einer ist, an welchem die Linie schrittweise ansteigt/abfällt, an die Positions-Berechnungseinheit 746. Ferner führt die Einheit 741 die Linien-Gradientenrichtungs-Code-Information DIR0 und DIR1 der Bestimmungseinheit 747 zu.

Ferner legt die Einheit 741 CC0- und CC1-Signale an die Berechnungseinheit 745 an, legt die RUAS0- und RUAS1-Signale an die den oberen Bereich erkennende Einheit 742U und die den rechten Bereich erkennende Einheit 742R der Peripherbereich-Erkennungseinrichtung 742 an und legt die LLAS0- und LLAS1-Signale an die den unteren Bereich erkennende Einheit 742D und die den linken Bereich erkennende Einheit 772L der Peripherbereich-Erkennungseinheit 742 an. Die zugeführten Signale wählen entsprechende Unterbereiche aus den drei Unterbereichen der jeweiligen Bereiche aus.

Die den oberen Bereich erkennende Einheit 742U, die den rechten Bereich erkennende Einheit 742R, die den unteren Bereich erkennende Einheit 742D und die den linken Bereich erkennende Einheit 742L der Peripherbereich-Erkennungseinheit 742 extrahiert annähernd die Daten der Punkte, welche in den ausgewählten Unterbereichen des oberen Bereichs 73U, des rechten Bereichs 73R, des unteren Bereichs 73D und des linken Bereichs 73L des Fensters 73 vorhanden sind. Folglich erkennt die Einheit 742 das relevante Linienmuster, um dem Multiplexer 743 oder 744 die "cn0 bis cn2", "dir0- und dir1"-Signale zu, welche die Anzahl serieller Punkte und die Gradientenrichtung der Linie anzeigen, die in den relevanten Unterbereichen vorhanden ist.

Der Multiplexer 743 wählt aus, um das Signal einzulegen,

geben, das von der den unteren Bereich erkennenden Einheit 742U zugeführt worden ist, wenn das Signal H/V auf "0" ist, und wählt das Signal aus, das von der den rechten Bereich erkennenden Einheit 742R zugeführt wird, wenn das Signal H/V auf "1" ist. Der Multiplexer 743 liefert dann die eingegebenen Signale, welche die Anzahl von seriellen Punkten in dem relevanten Unterbereich anzeigen, an die Gradienten-Berechnungseinheit 745 als die Signale RUCN0 bis RUCN2. Der Multiplexer 743 führt die eingegebenen Signale, welche die Gradientenrichtung der Linie in dem relevanten Unterbereich anzeigen, der Bestimmungseinheit 747 als die Signale RUDIR0 und RUDIR1 zu.

In ähnlicher Weise wählt der Multiplexer 744 aus, um das Signal einzugeben, das von der den unteren Bereich erkennenden Einheit 742D zugeführt worden ist, wenn das Signal H/V auf "0" ist, und wählt das Signal aus, das von der den linken Bereich erkennenden Einheit 742L zugeführt worden ist, wenn das Signal H/V auf "1" ist. Der Multiplexer 744 führt dann die eingegebenen Signale, welche die Anzahl an seriellen Daten in dem relevanten Unterbereich anzeigen, der Gradienten-Berechnungseinheit 745 und der Positions-Berechnungseinheit 746 als die Signale LLCN0 bis LLCN2 zu. Der Multiplexer 744 legt die eingegebenen Signale, welche die Gradientenrichtung der Linie in dem relevanten Unterbereich anzeigen, an die Bestimmungseinheit 747 als die Signale LLDIR und LLDIR1 an.

Die Bestimmungseinheit 747 bestimmt mit Hilfe der zugeführten Code-Information DIR0 und DIR1, RUDIR0 und RUDIR1, LLDIR0 und LLDIR1, ob die relevante Linie in der Zackenkorrekturoperation korrigiert werden sollte oder nicht. Wenn die Einheit 747 festlegt, die Linie zu korrigieren, liefert die Einheit 747 die Signale DIR0 und DIR1 und setzt "0" bei dem Signal "NICHT-AN-PASSEN". Wenn "1" bei dem NICHT-ANPASSEN gesetzt wird, schließt dieses Signal das Verknüpfungsglied 748, um zu verhindern, daß wichtige Positionsinformation P0 bis P3 zugeführt wird. Die Gradienten-Berechnungseinheit 745 berechnet den Gradienten (GRA-DIENT) des gerade erkannten Linienmusters als die Anzahl serieller Punkt, wie vorstehend beschrieben ist und liefert die entsprechende Codeinformation G0 bis G3. Bei dieser Berechnung wird die zugeführte Codeinformation CC0 und CC1, RUCN0 bis RUCN2, LLCN0 bis LLCN2 und die Signale RUC und LLC zugeführt.

Die Positions-Berechnungseinheit 746, welche die zugeführte Codeinformation LLCN0 bis LLCN2 und das Signal GST verwendet, berechnet die Position (POSITION) des relevanten Punktes und liefert die entsprechende Codeinformation p0 bis p3 (= P0 bis P3).

Verfahren zum Berechnen von Gradienten (GRA-DIENTS) durch die Gradienten-Berechnungseinheit 745 und zum Berechnen von Positionen (POSITIONS) durch die Positions-Berechnungseinheit 746 werden nunmehr beschrieben. Die Gradienten und Positionen werden mit Hilfe des GST- (der Wert (1-GST) wird nachstehend als "notGST" bezeichnet). Mit Hilfe von CC0 und CC1 (was nachstehend als CC bezeichnet wird), mit RUC und LLC, die von der Kernbereich-Erkennungseinheit 741 zugeführt worden sind, und mit RUCN0 bis RUCN2 (was als RUCN nachstehend bezeichnet wird), und mit LLCN0 bis LLCN2 (was nachstehend als LLCN bezeichnet wird), durch die folgenden Gleichungen (1) und (2):

$$\text{GRA-DIENT} = \text{CC} + \text{RUC} \cdot \text{RUCN} + \text{LLC} \cdot \text{LLCN} \quad (1);$$

$$\text{POSITION} = \text{GST} + \text{notGST} \cdot (\text{LLCN} + 2) \quad (2).$$

Konkrete Berechnungsbeispiele werden nunmehr für durch Punkte dargestellte Linienmusterbeispiele beschrieben, die in Fig. 39 bis 41 dargestellt sind. In jedem Beispiel ist ein relevanter Punkt einer, der in der d-ten Zeile und der sechsten Spalte in den  $(7 \times 11)$  Matrixpunkten angeordnet ist.

Nunmehr wird das Beispiel von Fig. 39 beschrieben. Der relevante Punkt ist nicht der Punkt, an welchem die Linie (welche durch die schwarz ausgelegten Kreise dargestellt ist) schrittweise ansteigt/abfällt. In dem Kernbereich 73C ist die Anzahl an seriellen Punkten 3 und in diesem Fall ist es notwendig, die beidseitigen peripheren Bereiche, d. h. den rechten Bereich 73R und den linken Bereich 73L zu überprüfen, und zwar deswegen, da es keine Höhendifferenz gibt, die auf der Linie in dem Kernbereich 73C vorhanden ist. Wenn es eine Höhendifferenz in dem Kernbereich 73C gibt, wie in Fig. 37 dargestellt, kann ein Überprüfen eines einseitigen peripheren Bereichs aus dem vorstehend beschriebenen Grund bei Fig. 37 entfallen. Folglich liefert die 20 Kernbereich erkennende Einheit 741 GST=0, CC=3, RUC=1 und LLCN=1.

Bezüglich der peripheren Bereiche 73R und 83L ist die Anzahl serieller Punkte, welche auf die aus relevanten Punkten gebildete Linie in dem Kernbereich 73C vorhanden ist, bevor ein Höhenunterschied auf der Linie erscheint, "1" in jedem Bereich, wie in der Figur dargestellt ist. Folglich liefern die MUx 743 und 744 RUCN=1 und LLCN=1.

Durch Substituieren der aktuellen Werte für die jeweiligen Terme der vorstehenden Gleichungen (1) und (2) ergibt sich:

$$\text{GRADIENT} = 3 + 1 \cdot 1 + 1 \cdot 1 = 3 + 1 + 1 = 5; \text{ und}$$

$$\text{POSITION} = 0 + (1 - 0) \cdot (1 + 2) = 0 + 1 \cdot 3 = 3$$

Nunmehr wird das Beispiel der Fig. 40 beschrieben. Das Beispiel in Fig. 40 enthält ein Linienmuster, das durch Verschieben eines Bits in dem Beispiel der Fig. 39 um ein Bit nach rechts erhalten worden ist. Folglich ist die Anzahl an seriellen Bits, die in dem peripheren Bereich 73R vorhanden ist, anschließend an die aus relevanten Punkten hergestellte Linie, die in dem Kernbereich 73C vorhanden ist, 2 und die Zahl, die in dem peripheren Bereich 73L vorhanden ist, ist 0. Folglich gilt: RUCN=2 und LLCN=0. Die andere Situation und folglich die sich ergebenden Werte sind dieselben wie bei dem Beispiel der Fig. 39.

Folglich ergibt sich durch Einsetzen der relevanten aktuellen Werte für die jeweiligen Terme der vorstehenden – Gleichung (1) und (2):

$$\text{GRADIENT} = 3 + 1 \cdot 2 + 1 \cdot 0 = 3 + 2 + 0 = 5 \text{ und}$$

$$\text{POSITION} = 0 + (1 - 0) \cdot (0 + 2) = 0 + 1 \cdot 2 = 2$$

Nunmehr wird das Beispiel der Fig. 41 beschrieben. Das Beispiel in Fig. 41 enthält ein Linienmuster, das dadurch erhalten worden ist, daß jedes Bit in dem Beispiel der Fig. 40 um ein Bit weiter nach rechts verschoben wird. Der relevante Punkt ist der Punkt, an welchem die Linie schrittweise ansteigt/abfällt, d. h. die Höhendifferenz ist in dem Kernbereich 63C vorhanden. In dem Kernbereich 73C ist die Anzahl an seriellen Punkten 2 und in diesem Fall muß nur der rechte periphere Bereich 73L aus dem Grund überprüft werden, der vorstehend bezüglich der Fig. 37 angeführt ist. Folglich liefert die Kernbereich-Erkennungseinheit 741 die GST=1, CC=2, RUC=1 und LLC=0.

Die Anzahl serieller Punkte, die anschließend an die aus relevanten Punkten hergestellte Zeile in dem Kernbereich vorhanden ist, bevor eine Höhendifferenz auf der Linie erscheint, ist für den rechten Bereich 73R 3 und für den linken Bereich 73L 4, wie in Fig. 41 dargestellt ist. Folglich liefern die MUx 743 und 744 RUCN=3 und LLCN=4.

Durch Einsetzen der vorstehenden aktuellen Werte für die jeweiligen Terme der obigen Gleichung (1) und (2) gilt:

$$\text{GRADIENT} = 2 + 1 \cdot 3 + 0 \cdot 4 = 2 + 3 + 0 = 5 \text{ und}$$

$$\text{POSITION} = 1 + (1 - 1) \cdot (4 + 2) = 1 + 0 \cdot 6 = 1.$$

Somit sind die Berechnungsbeispiele für Fälle von annähernd horizontalen Linienmustern beschrieben. Für Fälle von annähernd vertikalen Linienmustern werden die Gleichung (1) und (2) verwendet, um so entsprechende GRADIENTS und POSITIONS in ähnlicher Weise wie in den vorstehend beschriebenen Fällen für die annähernd horizontale Linienmuster zu erhalten. Jedoch wird in jedem Fall der annähernd vertikalen Linienmuster die Anzahl von seriellen Punkten, die in dem oberen peripheren Bereich 73U vorhanden sind, welcher an die aus relevanten Punkten gebildete Linie anschließt, die in dem Kernbereich 73C vorhanden ist, wird durch den Term RUCN ersetzt, und die Zahl, die in dem unteren peripheren Bereich 73D vorhanden ist, wird durch den Term LLCN ersetzt.

Nunmehr werden Verfahren zum Durchführen der Zacken-Korrekturoperation beschrieben. Die Zacken-Korrekturoperation, die an einer annähernd horizontalen Linie durchzuführen ist, wird nunmehr anhand von Fig. 37, 42A, 42B, 44A bis 44D beschrieben.

In Fig. 42A wird eine Punktinformation in einem Beispiel gemäß den eingegebenen Videodaten, welche von der Steuereinheit 3 angelegt worden sind, durch gestrichelt wiedergegebene Kreise in einem  $(7 \times 11)$  Matrix-Videobereich angezeigt. Die Punktinformation gemäß den eingegebenen Videodaten zeigt nur schwarz oder weiß für jeden Punkt und nicht die Größe des Punktes an. Folglich kann gesagt werden, daß die Größe jedes Punktes festgelegt ist. In Fig. 42A zeigen die schraffiert ausgelegten Kreise die Punkte an, die als ein Ergebnis der Zacken-Korrektur erhalten worden sind, die durchgeführt worden ist. Bei der Zacken-Korrekturoperation sind die Punktdurchmesser geändert worden und neue Punkte hinzugefügt worden, wie in Fig. 42A dargestellt worden ist. Die Größen/Durchmesser der Punkte entsprechen den Pulsbreiten, wenn beispielsweise der Laser der in Fig. 1 dargestellten Schreibeinheit 26 eingeschaltet ist. Die relevante Linie, welcher der Punktinformation entsprechend den von der Steuereinheit 3 zugeführten Videodaten entspricht, hat den Gradienten mit einer Neigung 1/5, was einer annähernd horizontalen Linie entspricht, wie in der Figur dargestellt ist. Folglich sind in dem  $(7 \times 11)$  Matrixbereich zwei Zacken (Höhendifferenzen) vorhanden, da jede Zacke als 5 Punkte-Intervall erscheint. Fig. 42B zeigt eine Pulsbreitenänderung beim Lasereinschalten, welche beim Durchführen der Zacken-Korrekturoperation an der d-ten Linie/Zeile des  $(7 \times 11)$  Matrixbereichs erhalten worden ist.

Das Beispiel der Fig. 37 entspricht einem, das das Fenster 73 enthält, wenn der Punkt in der d-ten Zeile und der neunten Spalte in der in Fig. 42A dargestellte  $(7 \times 11)$  Matrix angeordnet ist. Die Spalten (Fig. 37) in Fig. 44A bis 44D zeigen die in Fig. 9 dargestellten Signale in dem vorstehenden Fall. Fig. 44A zeigt Informa-

tionen, welche durch die Signale angezeigt ist, die von der Kernbereich-Erkennungseinheit 741 zugeführt werden sind, und Fig. 44B zeigt Information, welche durch die Signale angezeigt ist, die von der Peripherbereich-Erkennungseinheit 742 zugeführt sind. Fig. 44C zeigt in GRADIENT-Information, welche durch die Signale G0 bis G3 angezeigt ist, und Information, welche durch das Signal NICHT-ANPASSUNG angezeigt ist. Fig. 44 zeigt POSITION-Information, welche durch die Signale P1 bis P3 angezeigt ist.

Von den vorstehend wiedergegebenen Signalen werden die Signale H/V, DIR1, DIR0, B/W, U/L, G0 bis G3, P0 bis P3 als die Adresse im Musterspeicher 752 in dem in Fig. 4 dargestellten Speicher 75 verwendet. Folglich werden die Korrekturdaten aus dem Speicher-Muster 752 entsprechend der vorstehenden Adresse ausgelesen, wobei die Korrekturdaten die entsprechenden Videodaten sind. Die Videodaten werden kann an die Ansteuereinheit 4 gesendet, um den Laser in der in Fig. 1 dargestellten Schreibeinheit 26 anzusteuern.

Somit wird die Lasereinschalt-Pulsbreite, die zum Schreiben/Drucken des Punktes verwendet ist, der in der d-ten Zeile und der neunten Spalte in der in Fig. 42A dargestellten ( $7 \times 11$ ) Matrix angeordnet ist, auf 6/10 der vollen Pulsbreite reduziert, die für jeden Punkt vorher eingestellt worden ist, wie in Fig. 42B dargestellt ist. Folglich wird der relevante Punkte von dem durch eine gestrichelte Linie wiedergegebenen Kreis in den schraffierten Kreis geändert, wie in Fig. 42A dargestellt ist. Folglich wird der Punkt durchmesser auf 6/10 des vollen Punkt durchmessers verkleinert. Jeder der übrigen Punkt wird der relevante Punkt, so daß die entsprechenden Signale dem Musterspeicher zugeführt werden. Somit werden die entsprechenden Videodaten als die Korrekturdaten der Antriebsansteuereinheit 4 zugeführt. Somit sind die Punkte annähernd korrigiert, so daß die Punkte, welche als die schraffiert wiedergegebenen, in Fig. 42A dargestellten Kreise angezeigt sind, erhalten werden.

Selbst wenn ein relevanter Punkt ein weißer Punkt entsprechend Videodaten ist, die von der Steuereinheit 3 zugeführt worden sind, wird ein Punkt mit dem entsprechenden Durchmesser erzeugt, wenn es für den relevanten Punkte als ein Ergebnis des Linienmusters notwendig ist, das um den relevanten Punkte vorhanden ist, der erkannt worden ist. In den Ausführungsformen der Erfindung wird eine solche Punkt durchmesser-(Lasereinschalt-Pulsbreiten-)Verkleinerung durchgeführt, so daß ein bestimmter Wert als eine Einheit verwendet wird, welche für die vorstehend erwähnte Verkleinerung zu verwenden ist. Der spezielle Wert ist einer, welcher dadurch erhalten wird, daß 1 durch eine ganz bestimmte ganze Zahl geteilt wird. In dem vorstehend wiedergegebenen Beispiel ist der spezielle Wert 1/10.

Bei Fig. 42A scheint es, daß Risse zwischen benachbarten Punktlinien infolge der Punkt durchmesser-Verkleinerung vorhanden sind, welche an den zwei Enden jeder Punktlinie durchgeführt worden ist. Jedoch bei einem tatsächlichen Druckprozeß mittels eines Laserdruckers ist die Druckauflösung nicht so fein, daß Unschärfe erscheinen. Folglich werden die benachbarten Punktlinien durch den Druckprozeß in einer einzigen Linie integriert, folglich sind die Zacken, welche in der Punktinformation gemäß den von der Steuereinheit 3 zugeführten Videodaten vorhanden sind, entsprechend korrigiert, so daß die leicht schräge, glatte gerade Linie erhalten wird.

Das Beispiel in Fig. 42A ist: eines für eine Linie, die die

Breite eines einzigen Punktes hat, c. h. für eine sehr dünne Linie in einem Bild. Wenn angenommen wird, daß eine zu verarbeitende/korrigierende Linie eine Grenzlinie zwischen einem schwarz ausgelegten (Schwarzpunkt-)Bereich und einem weißen (Weißpunkt-)Bereich ist, wird die Zacken-Korrekturoperation entsprechend geändert, wie nachstehend beschrieben wird. Mit anderen Worten, die vorstehend wiedergegebene Annahme ist die, daß es einen Schwarzpunkt-Bereich mit einer Breite gibt, die breiter als diejenige eines einzelnen Punktes ist. Wenn in einem solchen angenommenen Fall die Zacken-Korrektur durchgeführt wird, werden eine Punkt-Durchmesser-Verkleinerung und ein Punkt-Hinzufügen am Ende der Punktlinie durchgeführt, und zwar an dem Ende, an welchem die relevante Linie an einer Kante des Schwarzpunkt-Bereichs angeordnet ist. Jedoch wird keine Punkt Durchmesser-Verkleinerung an dem Ende der Punktlinie durchgeführt, an welchem die relevante Punktlinie sich in dem Inneren des Schwarzpunkt-Bereichs fortsetzt. Am Ende der Punktlinie muß folglich nicht notwendigerweise das Punkt-Hinzufügen durchgeführt werden.

Diese Möglichkeiten werden nunmehr in Verbindung mit dem Beispiel in Fig. 42A beschrieben. Es wird angenommen, daß der gesamte Bereich, welcher unter den Punktlinien vorhanden ist, die in der in Fig. 42A dargestellten ( $7 \times 11$ ) Matrix vorhanden sind, der Schwarzpunkt-Bereich ist. D.h. es wird angenommen, daß das in Fig. 42A dargestellte Punktbild eines ist, das einer Grenzlinie zwischen dem oberen Weißpunkt-Bereich und dem unteren Schwarzpunkt-Bereich entspricht, wobei die Linie die schräge gerade Linie mit einer Neigung von 1/5 ist. In diesem angenommenen Fall werden die schwarzen Punkte, die in der e-ten Zeile, den zweiten und dritten Spalten der d-ten Zeile, den siebten und achten Spalten angeordnet sind, als volle Punkte belassen, ohne daß irgendeine Punkt-Durchmesser-Verkleinerung durchgeführt wird. Ferner wird kein Punkt in der e-ten Zeile, den vierten und fünften Spalten, der d-ten Zeile und den neunten und A-ten Spalten hinzugefügt, da diese Positionen bei der vorstehend wiedergegebenen Annahme bereits von den vollen schwarzen Punkten besetzt sind.

Die Zacken-Korrekturoperation, die an einer annähernd vertikalen Linie durchzuführen ist, wird nunmehr anhand von Fig. 38, 43A, 43B, sowie 44A bis 44D durchgeführt.

In Fig. 43A ist beispielsweise eine Punktinformation gemäß den eingegebenen Videodaten, die von der Steuereinheit 3 zugeführt worden sind, durch von gestrichelten Linien eingeschlossenen Kreisen in einem ( $7 \times 11$ ) Matrix-Videobereich dargestellt. Die Punktinformation gemäß den eingegebenen Videodaten zeigt nur schwarz oder weiß für jeden Punkt und zeigt nicht die Größe des Punktes an, wie vorstehend beschrieben ist. In Fig. 43A zeigen die schraffierte ausgelegten Kreise die Punkte an, welche als ein Ergebnis der Zacken-Korrektur erhalten worden sind, die an diesen durchgeführt worden ist. Bei der Zacken-Korrekturoperation sind die Positionen der Punkte entsprechend geändert worden, wie in Fig. 43A dargestellt ist. Eine solche Punktpositionsänderung entspricht der Änderung in dem Erscheinungszeitpunkt von Impulsen, wenn beispielsweise der Laser der in Fig. 1 dargestellten Schreibeinheit 26 angeschaltet wird. Die relevante Linie, welche der Punktinformation gemäß den von der Steuereinheit 3 zugeführten Daten entspricht, hat einen Gradienten mit einer Neigung 3/1, was einer annähernd vertikalen Linie ent-

spricht, wie in Fig. 43A dargestellt ist. Folglich sind in dem  $(7 \times 11)$  Matrixbereich zwei Zacken (Unterschiede in Positionen entlang der horizontalen Richtung) vorhanden, wenn jede Zacke in Dreipunkt-Intervallen erscheint. Fig. 43B zeigt eine zeitliche Pulsbreiten-Verschiebung beim Laser-Einschalten, welche beim Durchführen der Zacken-Korrekturoperation auf der b-ten Zeile des  $(7 \times 11)$  Matrixbereich erhalten worden ist.

Das Beispiel der Fig. 38 entspricht einem, welches das Fenster 73 in dem Fall enthält, daß der Punkt in der b-ten Zeile und der fünften Spalte in der in Fig. 43 dargestellten  $(7 \times 11)$  Matrix angeordnet ist. Die Spalten (38) in Fig. 44A bis 44 D zeigen die in Fig. 9 dargestellten Signale in dem oben angeführten Fall.

Von diesen Signalen werden die Signale H/V, DIR1, DIR0, B/W, U/L, G3 bis G0, P3 bis P0 als die Adresse in dem Musterspeicher 752 in dem in Fig. 4 dargestellten Musterblock verwendet, wie vorstehend beschrieben ist. Folglich werden die Korrekturdaten aus dem Musterspeicher 752 entsprechend der obigen Adresse ausgelesen, wobei die Korrekturdaten die entsprechenden Videodaten sind. Die Videodaten werden dann an die Antriebs-Ansteuereinheit 4 abgegeben, um so den Laser in der in Fig. 1 dargestellten Schreibeinheit 26 anzusteuern.

Folglich ist der Erscheinungszeitpunkt des Laser-Anschaltimpulses, der beim Schreiben/Drucken des Punktes verwendet worden ist, der in der b-ten Zeile und der fünften Spalte der in Fig. 43A dargestellten  $(7 \times 11)$  Matrix angeordnet ist, um  $1/3$  des Zeitabschnitts verzögert, welcher der vollen Pulsbreite ohne irgendeine Änderung entspricht, welche in der Pulsbreite vorkommt, wie in Fig. 43B dargestellt ist. Folglich wird die Position des relevanten Punktes von der Position des durch eine gestrichelte Linie wiedergegebenen Kreises nach rechts in die Position des schraffiert wiedergegebenes Kreises durch  $1/3$  des Durchmesser des Punktes geändert, wie in Fig. 42A dargestellt ist. Folglich wird der Punkt-Durchmesser auf  $6/10$  des vollen Punkt-Durchmessers verkleinert. Jeder der übrigen Punkte wird der relevante Punkt, so daß die entsprechenden Signale dem Musterspeicher zugeführt werden. Folglich werden die entsprechenden Videodaten als die Korrekturdaten der Antriebs-Ansteuereinheit 4 zugefügt. Somit sind die Punkt-Positionen dementsprechend in horizontaler Richtung geändert, so daß die Punkte, welche als die in Fig. 43A dargestellten, schraffierten Kreise wiedergegeben sind, erhalten werden. Folglich sind die Zacken, die in der Punktinformation entsprechend den von der Steuereinheit 3 zugeführten Videodaten vorhanden sind, annähernd korrigiert, so daß die leicht schräge, glatte, gerade Linie erhalten wird. Ähnlich wie bei dem vorstehend beschriebenen Fall wird in den Ausführungsformen der Erfindung eine solche Punkt-Positions-(Lasereinschalt-Impulszeitpunkt-)Verschiebung so durchgeführt, daß ein bestimmter Wert als eine Einheit verwendet wird, welche für die Verschiebung zu verwenden ist. Der bestimmte Wert ist einer, welcher durch Teilen von eins durch eine bestimmte ganze Zahl erhalten worden ist. In dem vorstehenden Beispiel ist der bestimmte Wert  $1/10$ .

Das vorstehende Beispiel der Fig. 43A ist eines für eine Linie, welche die Breite eines einzigen Punktes hat, d. h. für eine sehr dünne Linie in einem Bildpunkt. Wenn angenommen wird, daß eine zu verarbeitende/korrigierende Linie eine Grenzlinie zwischen einem schwarz ausgelegten (Schwarzpunkt-)Bereich und einem weißen (Weißpunkt-)Bereich ist, wird die Zacken-Korrekturoperation entsprechend geändert, wie nachstehend be-

schrieben wird. Mit anderen Worten, die vorstehende Annahme ist die, daß ein Schwarzpunkt-Bereich mit einer Breite, die größer als diejenige eines einzelnen Punktes ist. Wenn in einem solchen angenommenen Fall die Punkt-Positionsverschiebung bei der Zacken-Korrekturoperation gefordert wird, so daß eine Punktposition um eine bestimmte Strecke von der Schwarzpunkt-Bereichsseite zu der Weißpunkt-Bereichsseite verschoben wird, wird der Punkt unverschoben gelassen und ein anderer Punkt mit einer Position, die aus der Position des ursprünglichen Punktes um die bestimmte Strecke verschoben worden ist, wird hinzugefügt.

Diese Möglichkeit wird nunmehr anhand des Beispiels der Fig. 43A beschrieben. Es ist angenommen, daß der gesamte Bereich, der auf der linken Seite der Punkt-Linie vorhanden ist, die in der in Fig. 43A dargestellten  $(7 \times 11)$  Matrix vorhanden ist, der Schwarzpunktbereich. D.h. es wird angenommen, daß das in Fig. 43A dargestellte Punktbild eines ist, da einer Grenzlinie zwischen dem rechten Weißpunkt-Bereich und dem linken Schwarzpunkt-Bereich entspricht, wobei die Linie die schräge, gerade Linie mit einer Neigung von  $3/1$  ist. In diesem angenommenen Fall werden die schwarzen Punkte, die in der b-ten Zeile, der fünften Spalte und der e-ten Zeile sowie der sechsten Spalte angeordnet sind, in den Positionen unverschoben belassen, welche durch die gestrichelt wiedergegebenen Kreise angezeigt sind, und andere Punkte werden hinzugefügt. Diese anderen Punkte sind durch schraffierte Kreise dargestellt und haben die Positionen, die (zu der Weißpunkt-Bereichsseite bei der vorstehenden Annahme) aus der Position nach rechts verschoben sind, welche durch die gestrichelt wiedergegebenen Kreise angezeigt ist. Bei dieser Annahme werden die schwarzen Punkte, die in der c-ten Zeile, der sechsten Spalte und der f-ten Zeile sowie der siebten Spalte angeordnet sind, nach links (zu der Schwarzpunkt-Bereichsseite) um  $1/3$  des vollen Durchmessers aus den durch gestrichelte Linien wiedergegebenen Kreispositionen in die schraffiert ausgefüllten Kreispositionen verschoben. Folglich gibt es ein Überdecken von zwei Punkten in dem Schwarzpunkt-Bereich. Ein derartiges Punktüberdecken entspricht zwei Laser-Einschaltimpulsen, die sich überdecken, um dadurch einen kontinuierlichen Doppelimpuls zu bilden.

Obwohl die Punktkorrekturseinheit 7, die als das Punktverarbeitungssystem gemäß der Erfindung wirkt, in dem internen Interface 5 vorgesehen ist, das die Steuereinheit 3 mit der Ansteuereinheit 4 in dem Laser-Drucker 2 in den vorstehend beschriebenen Ausführungsformen verbindet, ist es im Rahmen der Erfindung auch möglich, die Punkt-Korrekturseinheit 7 in der Steuereinheit 3 oder in der Antriebs-Ansteuereinheit 4 vorgesehen.

Ferner kann die Erfindung nicht nur bei einem solchen Laser-Drucker, sondern auch bei verschiedenen Bilderzeugungseinrichtungen und bei Bildanzeige/Druckeinrichtungen zum Darstellen/Drucken von Bildern angewendet werden, welche durch die vorstehend beschriebene Bilderzeugungseinrichtung erzeugt worden sind. Diese Einrichtungen sind beispielsweise verschiedene optische Drucker/Printer, wie in LED-Drucker/Drucker, ein Digital-Kopierer, ein Faksimilegerät usw. Diese Einrichtungen erzeugen Bilder, indem eingegebene Videodaten in die Bitmap-Formationsdaten gedehnt werden und die auf diese Weise erzeugten Bilder dargestellt/gedruckt werden.

## Zu Fig. 1

- 3 Steuereinheit
- 35 Bedienungsfeld
- 7 Punkt-Korrektur-Einheit
- 4 Antriebs-Ansteuereinheit
- 5 Printer-Antrieb
- 26 Schreibeinheit
- 27 Folge-Gruppe
- 28 Sensorarten

## Zu Fig. 4

## (Von Steuereinheit 3)

- 73 Fenstereinheit (Schieberegister)
- 74 Muster - Erkennungseinheit
- 75 Speicherblock
- 76 Videodaten-Ausgabeeinheit

## (Von Antriebs-Steuereinheit 4)

## (Zu Steuereinheit 3)

- 78 Steuersignal erzeugende Einrichtung

## Zu Fig. 5

- Von Ausgang 72a, 72b, 72c, 72d, 72e, 72f 72a bis 72g
- Zeilenpuffer
- 72a-, 72b-, 72c-, 72d-, 72e-, 72f-Ausgang
- 74 Muster-Erkennungseinheit

## Zu Fig. 6C bis Fig. 6I und zu Fig. 7C bis 7I

- 72a-, 72b-, 72c-, 72d-, 72e-, 72f- und 72g-Ausgang

## Zu Fig. 9

- 741 Kernbereich- Erkennungseinheit
- 742 Peripherbereich- Erkennungseinrichtung
- 742U Obere
- 742R Rechte
- 742D Untere
- 742L Linke
- 73U Oberer
- 73L Linker
- 73R Rechter
- 73C Kern(3 x 3)
- 73D Unterer
- 774 Bestimmungseinheit:
- 745 Gradienten-Berechnungseinheit
- 746 Positions-Berechnungseinheit

## Zu Fig. 10 bis 13

- 74 Muster-Erkennungseinheit
- 752 Musterspeicher
- 751 Tabellenspeicher

## Zu Fig. 14

- 1 Ausführungsform
- 2 Tabellenspeicher-Kapazität (A)
- 3 Musterspeicher-Kapazität (B)
- 4 Gesamtkapazität (A + B)

## Zu Fig. 15

- 74 Muster-Erkennungseinheit
- 90 Codeinformations-Schalteinrichtung
- 75 Speicherblock

## Zu Fig. 17A bis 17F

- 1 Videodaten von Steuereinheit 3
- 2 Bildkorrektur- Ergebnis
- 5 3 Erste Zeile
- 4 Zweite Zeile
- 5 Dritte Zeile
- 6 Vierte Zeile
- 7 Dieser Musterteil muß korrigiert werden, wie in
- 10 Fig. 17F dargestellt
- 8 Bildmuster
- 9 Vergrößerung von gestricheltem Kreisbereich
- 10 FIFO-Speicher 72
- Ausgabedaten
- 15 11 A12 bis A15 ist verwendet worden
- 12 RA12 bis RA15 ist verwendet worden
- 13 Korrekturbispiel 1
- 14 Korrekturbispiel 2

## Zu Fig. 18

- (Von Steuereinheit 3)
- 73 Fenstereinheit (Schieberegister)
- 74 Muster-Erkennungseinheit
- 25 75 Speicherblock
- 76 Videodaten-Ausgabeeinheit
- 80 Korrigierte Videodaten
- 77 Zeitsteuereinheit
- 78 Steuersignal erzeugende Einrichtung
- 30 (Zu Steuereinheit 3)
- (Von Antriebs-Ansteuereinheit 4)
- 79 Bilddaten-Handhabungseinrichtung
- 81 Behandelte Videodaten

## Zu Fig. 19 und 20

- (Von Steuereinheit 3)
- 70 Mustererkennungs-Verarbeitungseinheit
- 75 Speicherblock
- 40 76 Videodaten-Ausgabeeinheit
- 79 Bilddaten-Handhabungseinrichtung
- 80 Behandelte Daten zu Fig. 21

## Zu Fig. 21, Fig. 23, Fig. 25

- 45 (Von Steuereinheit 3)
- 73 Fenstereinheit (Schieberegister)
- 74 Muster-Erkennungseinheit
- 75 Speicherblock
- 50 76 Videodaten-Ausgabeeinheit
- (Zu Steuereinheit 3)
- 78 Steuersignal erzeugende Einrichtung
- (Von Antriebs-Ansteuereinheit 4)
- 80 Fensterbereich-Einstelleinrichtung
- 55 81 Fensterbereich-Bilddaten-Verarbeitungs-Einstell-einrichtung
- 82 Fensterprioritäts-Einstelleinrichtung
- 82a Bereicheinstelldaten (von CPU usw.)
- 83 Fensterbereich-Bilddaten-Verarbeitungsmethode-
- 60 Einstelleinrichtung

## Patentansprüche

- 65 1. Bilddaten-Verarbeitungssystem, gekennzeich-net durch eine Bitmap-Einrichtung (3) zum Erzeu-gen eines Bitmap mit Hilfe vorgegebener Bilddaten;
- eine Grenze erkennende Einrichtung (74), um

Kenndaten einer Grenze zu erkennen, die zwischen einer ersten Pixelart und einer zweiten Pixelart vorhanden sind, welche das Bitmap bilden; eine Bestimmungseinrichtung (74), um für jedes Pixel zu bestimmen, ob eine Korrektur notwendig ist, um so das aus den vorgegebenen Bilddaten erzeugte Bild deutlich feiner zu machen, wobei bei der Feststellung zumindest ein Teil der Grenz-Kenndaten verwendet wird; eine Korrektureinrichtung (75), um eine Korrektur an den Bilddaten für das Pixel durchzuführen, dessen Korrektur als notwendig festgestellt worden ist, wobei die Korrektur in einer Weise durchgeführt wird, die in Abhängigkeit von den relevanten Grenz-Kenndaten bestimmt wird, und eine Datenwiederholeinrichtung (72), um die vorgegebenen Bilddaten zu wiederholen, die bei der Korrektur zu verwenden sind, die von der Korrektureinrichtung durchzuführen ist.

2. Bilddaten-Verarbeitungssystem nach Anspruch 1, dadurch gekennzeichnet, daß das System ferner eine Einrichtung (78) zum Erzeugen einer zeitlichen Steuerung aufweist, bei welcher die Datenwiederholeinrichtung die Wiederholoperation durchführt.

3. Bilddaten-Verarbeitungssystem nach Anspruch 1, dadurch gekennzeichnet, daß das System ferner Einstelleinrichtungen (78) aufweist, um die Anzahl Mal einzustellen, wie oft die Datenwiederholeinrichtung die Wiederholoperation durchführt.

4. Bilddaten-Verarbeitungssystem nach Anspruch 1, dadurch gekennzeichnet, daß das System ferner eine Numerierungseinrichtung (78) aufweist, um den Bilddatensatz zu numerieren, der bei der Wiederholung für jede Wiederholoperation erzeugt worden ist, um so die Zahl jedes Bilddatensatzes zu identifizieren, der bei der Wiederholung erzeugt worden ist.

5. Bilddaten-Verarbeitungssystem nach Anspruch 4, dadurch gekennzeichnet, daß das System ferner aufweist:

eine Fenstereinrichtung (73), um einen vorherbestimmten Bereich von Pixels in dem Bitmap zu extrahieren, in welchem Bereich eine relevantes Pixel, das gerade zu verarbeiten ist, zentriert wird, und wobei die eine Grenze erkennende Einrichtung die Kenndaten der Grenze zwischen der ersten Pixelart und der zweiten Pixelart in dem Bereich erkennt, und die Korrektureinrichtung die Korrekturart als ein Ergebnis des Auswählens aus einer Anzahl vorherbestimmter Arten zuführt, welche in Abhängigkeit von verschiedenen Mustern der Grenz-Kenndaten und in Abhängigkeit von den Zahlen klassifiziert sind, welche durch die Numerierungseinrichtung für wiederholte Sätze von Bilddaten für jeden ursprünglichen Satz von Bilddaten festgelegt worden sind.

6. Bilddaten-Verarbeitungssystem nach Anspruch 4, dadurch gekennzeichnet, daß die Korrektureinrichtung einen Musterspeicher verwendet, in dem vorher eine Anzahl vorherbestimmter Korrekturmuster gespeichert wird, welche der Anzahl vorherbestimmter Arten entsprechen, wobei der Speicher so ausgebildet ist, daß eine entsprechende Art aus einer Anzahl vorherbestimmter Arten in Abhängigkeit von den verschiedenen Mustern der Grenz-Kenndaten und in Abhängigkeit von den Zahlen ausgewählt werden kann, welche durch die

Numerierungseinrichtung für wiederholte Sätze von Bilddaten für jeden ursprünglichen Satz von Bilddaten bestimmt werden ist.

7. Bilddaten-Verarbeitungssystem gemäß Anspruch 6, dadurch gekennzeichnet, daß der Musterspeicher einen ersten Speicher und einen zweiten Speicher enthält, wobei der zweite Speicher vorher die Anzahl vorherbestimmter Korrekturmuster speichert und der zweite Speicher so ausgewählt ist, daß ein Muster aus einer Anzahl vorherbestimmter Korrekturmuster mit Hilfe eines Codes ausgewählt werden kann, der aus vorher in dem ersten Speicher gespeicherter Codes ausgewählt ist, und wobei der erste Speicher vorher die Codes speichert und der erste Speicher so ausgeführt ist, daß der entsprechende Code aus den Codes in Abhängigkeit von den verschiedenen Mustern der Grenz-Kenndaten und in Abhängigkeit von den Zahlen ausgewählt werden kann, welche von der Numerierungseinrichtung für wiederholte Sätze von Bilddaten für jeden ursprünglichen Satz von Bilddaten festgelegt worden ist.

8. Bilddaten-Verarbeitungssystem nach Anspruch 6, dadurch gekennzeichnet, daß der Musterspeicher einen ersten Speicher und einen zweiten Speicher enthält, wobei der zweite Speicher vorher die Anzahl vorherbestimmter Korrekturmuster speichert und der zweite Speicher so ausgebildet ist, daß ein Muster aus der Anzahl vorherbestimmter Korrekturmuster mit Hilfe eines Codes ausgewählt werden kann, der aus vorher in dem ersten Speicher gespeicherten Codes und der relevanten Zahl und den Zahlen ausgewählt ist, welche durch die Numerierungseinrichtung für wiederholte Sätze von Bilddaten für jeden ursprünglichen Satz von Bilddaten ausgewählt ist, wobei der erste Speicher vorher die Codes speichert und der erste Speicher so ausgebildet ist, daß der entsprechende Code aus den Codes in Abhängigkeit von den verschiedenen Mustern der Grenz-Kenndaten ausgewählt werden kann.

9. Bilddaten-Verarbeitungssystem nach Anspruch 4, dadurch gekennzeichnet, daß das System ferner eine Umkehr-Numerierungseinrichtung (78) aufweist, um Zahlen zu erzeugen, die durch Umkehren der Numerierung erhalten worden sind, welche der Numerierungseinrichtung zugeführt worden ist.

10. Bilddaten-Verarbeitungssystem nach Anspruch 1, dadurch gekennzeichnet, daß das System ferner eine Lichtemissionsintensität-Steuereinrichtung (26) aufweist, um die Lichtemissionsintensität einer Laserdiode entsprechend der von der Korrektur einrichtung durchgeföhrten Korrektur zu steuern.

11. Bilddaten-Verarbeitungssystem nach Anspruch 1, dadurch gekennzeichnet, daß das System ferner eine Handhabungseinrichtung (79) aufweist, um die vorgegebenen Bilddaten beliebig zu handhaben, wobei die Handhabungseinrichtung entweder die Grenz-Kenndaten oder die Bilddaten verarbeitet, welche der von der Korrektureinrichtung durchgeföhrten Korrektur unterzogen werden.

12. Bilddaten-Verarbeitungssystem nach Anspruch 11, dadurch gekennzeichnet, daß die Handhabungseinrichtung einen Mikrocomputer aufweist.

13. Bilddaten-Verarbeitungssystem nach Anspruch 11, dadurch gekennzeichnet, daß die Korrektur an Bilddaten, die von der Handhabungseinrichtung erzeugt worden sind, mittels der Korrektureinrichtung durchgeföhrzt wird.

14. Bilddaten-Verarbeitungssystem nach Anspruch 11, dadurch gekennzeichnet, daß die Korrektureinrichtung und die Handhabungseinrichtung parallel arbeiten.
15. Bilddaten-Verarbeitungssystem nach Anspruch 1, dadurch gekennzeichnet, daß das System ferner aufweist:
- eine takterzeugende Einrichtung (78) im Innern des Systems, um interne Taktimpulse zu erzeugen, und eine Auswähleinrichtung (78), um Taktimpulse synchron mit den Impulsen, mit welchen jede Einrichtung betrieben wird, zwischen dem internen Taktimpulsen und den externen Taktimpulsen auszuwählen, welche von außerhalb des Systems angelegt werden können.
  - 16. Bilddaten-Verarbeitungssystem nach Anspruch 6, dadurch gekennzeichnet, daß die Zahl der Anzahl vorherbestimmter Muster so stark reduziert wird, daß die wesentliche Feinheit in dem zu verwirklichen Bild erhalten wird.
  - 17. Bilddaten-Verarbeitungssystem dadurch gekennzeichnet, daß das System aufweist:
    - eine Bitmap-Einrichtung (2), um ein Bitmap mit Hilfe vorgegebener Bilddaten zu erzeugen;
    - eine eine Grenze erkennende Einrichtung (74), um Kenndaten einer Grenze zu erkennen, welche zwischen ersten Pixelart und einer zweiten Pixelart vorhanden ist, welche das Bitmap bilden;
    - eine Bestimmungseinrichtung (74), um für jedes Pixel zu bestimmen, ob eine Korrektur notwendig ist, um das aus den vorgegebenen Bilddaten erzeugte Bild erheblich feiner zu machen, wobei bei der Bestimmung zumindest ein Teil der Grenz-Kenndaten verwendet wird;
    - eine Korrektur-Einrichtung (75), um eine Korrektur an den Bilddaten für das Pixel durchzuführen, dessen Korrektur als notwendig festgestellt worden ist, wobei die Korrektur in einer Weise durchgeführt wird, die in Abhängigkeit von den relevanten Grenz-Kenndaten bestimmt wird, und
    - eine Vielfenster-Einrichtung (80), um eine Anzahl Fenster auf einer Seite des Bitmaps festzulegen.

18. Bilddaten-Verarbeitungssystem nach Anspruch 17, dadurch gekennzeichnet, daß das System ferner eine Korrektur-Einstelleinrichtung (81) aufweist, um für jedes Fenster, das durch die Vielfenster-Einrichtung festgelegt worden ist, zu bestimmen, ob die Korrektur von der Korrektur-Einrichtung durchgeführt wird.

19. Bilddaten-Verarbeitungssystem nach Anspruch 17, dadurch gekennzeichnet, daß das System ferner eine Prioritäten-Sezeinrichtung (83) aufweist, um eine Fenster-Priorität zu bestimmen, in welcher die Korrekturen von der Korrektur-Einrichtung an den Bilddaten durchgeführt werden, welche durch eines der Anzahl der Fenster festgelegt worden sind.

20. Bilddaten-Verarbeitungssystem nach Anspruch 17, dadurch gekennzeichnet, daß das System ferner eine Spezialmethoden-Einstelleinrichtung (83) aufweist, um eine spezielle Methode auszuwählen, gemäß welcher die durch die relevanten Fenster festgelegten Bilddaten zusätzlich zu der mittels der Korrektur-Einrichtung durchgeföhrten Korrektur verarbeitet werden, wobei die spezielle Methode aus speziellen Methoden ausgewählt wird, welche verschiedene Arten von Bildverarbeitungen beschreiben.

21. Bilddaten-Verarbeitungssystem dadurch ge-

- kennzeichnet, daß das System aufweist:
- eine Datenwiederholeinrichtung (72), um ursprüngliche Bilddaten zu wiederholen, um so wiederholte Bilddaten zu erhalten;
  - eine Addiereinrichtung (72), um die ursprünglichen Daten und die wiederholten Daten miteinander hinzuzufügen, um revidierte Bilddaten zu erzeugen, und
  - eine Korrektur-Einrichtung (75), um die ursprünglichen Bilddaten und die wiederholten Bilddaten zu handhaben.
22. Bilddaten-Verarbeitungssystem dadurch gekennzeichnet, daß das System aufweist:
- eine Fenstereinrichtung (80), um zumindest ein Fenster in einem vorgegebenen Bild festzulegen, und
  - eine Behandlungseinrichtung (75, 81 und 83), um das vorgegebene Bild in einem durch das Fenster festgelegten Bereich zu handhaben.
23. Bilddaten-Verarbeitungsverfahren, dadurch gekennzeichnet, daß das Verfahren aufweist:
- a) einen Bitmap-Schritt, bei dem ein Bitmap mit Hilfe vorgegebener Daten gebildet wird;
  - b) einen die Grenze erkennenden Schritt, bei welcher Kenndaten einer Grenze zwischen einer ersten Pixelart und einer zweiten Pixelart erkannt werden, welche das Bitmap bilden;
  - c) einen Bestimmungsschritt, welcher für jedes Pixel bestimmt, ob eine Korrektur notwendig ist, um so das aus den vorgegebenen Bilddaten erzeugte Bild deutlich feiner zu machen, wobei für die Bestimmung zumindest ein Teil der Grenz-Kenndaten verwendet wird;
  - d) einen Korrekturschritt, bei welchem eine Korrektur an den Bilddaten für das Bildelement durchgeführt wird, deren Korrektur als notwendig festgelegt worden ist, wobei die Korrekturdaten in einer Weise durchgeführt werden, welche in Abhängigkeit von den relevanten Grenz-Kenndaten festgelegt worden ist, und
  - e) einen Datenwiederholsschritt, bei welchem die vorgegebenen Bilddaten wiederholt werden, die bei der Korrektur zu verwenden sind, die beim Korrekturschritt d) durchzuführen ist.
24. Bilddaten-Verarbeitungsverfahren, dadurch gekennzeichnet, daß das Verfahren aufweist:
- a) einen Bitmap-Schritt, bei welchem ein Bitmap mit Hilfe von vorgegebenen Bilddaten erzeugt wird;
  - b) einen eine Grenze erkennenden Schritt, bei welchem Kenndaten einer Grenze zwischen einer ersten Pixelart und einer zweiten Pixelart erkannt werden, welche das Bitmap darstellen;
  - c) einen Bestimmungsschritt, welcher für jedes Pixel bestimmt, ob eine Korrektur notwendig ist, um das aus den vorgegebenen Bilddaten erzeugte Bild deutlich feiner zu machen, wobei bei der Bestimmung zumindest ein Teil der Grenz-Kenndaten verwendet wird;
  - d) einen Korrekturschritt, um eine Korrektur an den Bilddaten für das Pixel durchzuführen, dessen Korrektur als notwendig festgestellt worden ist, wobei die Korrektur in einer Weise durchgeführt wird, welche in Abhängigkeit von den relevanten Grenz-Kenndaten be-

stimmt wird, und

e) einen Vielfenster-Schritt, bei welchem eine Anzahl Fenster auf einer Seite des Bitmap definiert wird.

25. Bilddaten-Verarbeitungsverfahren mit
- a) einem Datenwiederholschritt, bei welchem 5 ursprüngliche Bilddaten wiederholt werden, um so wiederholte Bilddaten zu erhalten;
  - b) einen Addierschritt, um die ursprünglichen Daten und die wiederholten Daten hinzuzufügen, um so revidierte Bilddaten zu erzeugen und
  - c) einen Korrigierschritt, bei welchem die 15 ursprünglichen Daten und die wiederholten Bilddaten behandelt werden.

26. Bilddaten-Verarbeitungssystem mit einer Fenstereinrichtung, um zumindest ein Fenster in einem vorgegebenen Bild festzulegen, und mit einer Handhabungseinrichtung, um das vorgegebene Bild in einem durch das Fenster festgelegten Bereich zu handhaben. 20

---

Hierzu 36 Seite(n) Zeichnungen

---

25

30

35

40

45

50

55

60

65

FIG. I

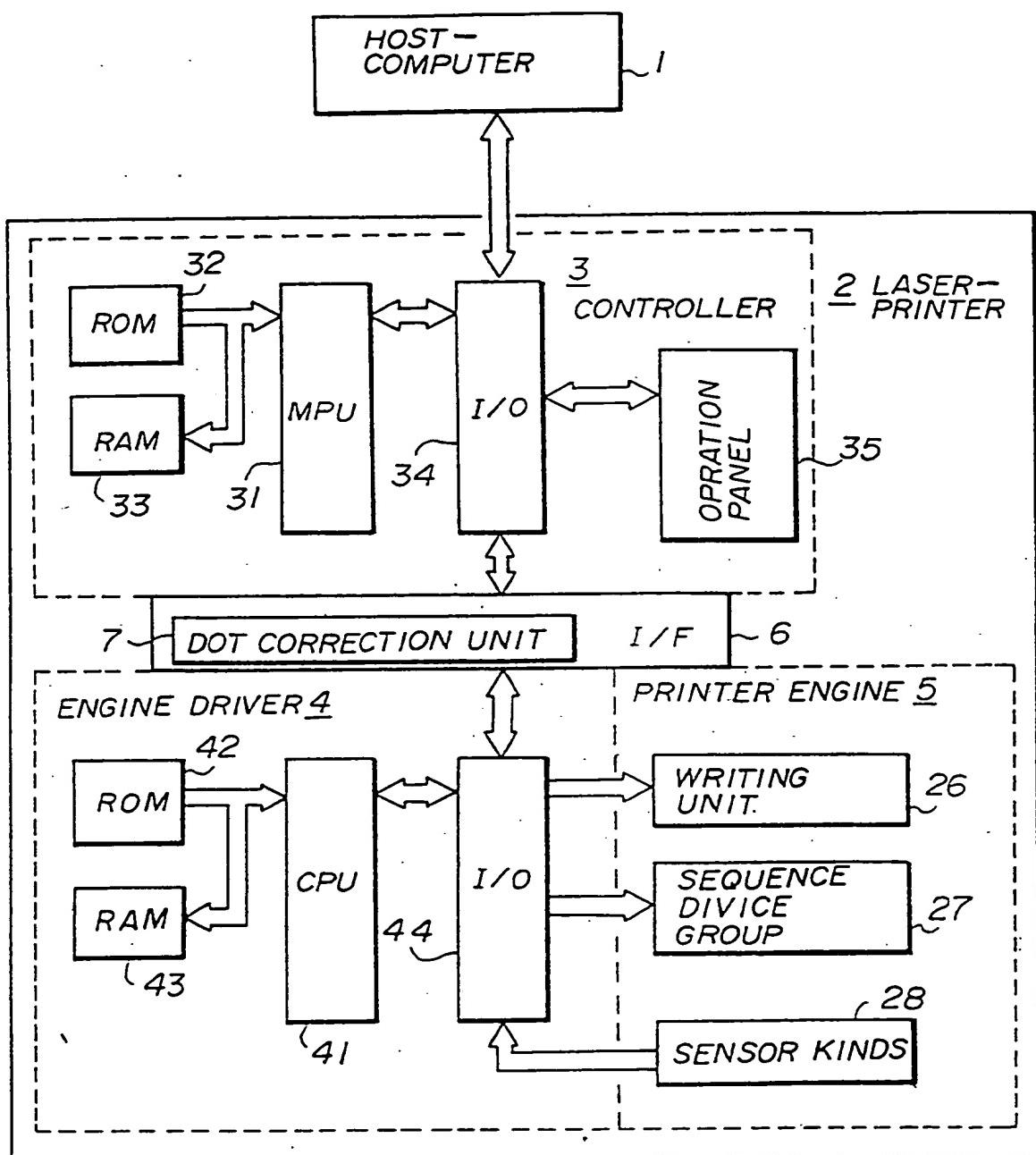


FIG. 2

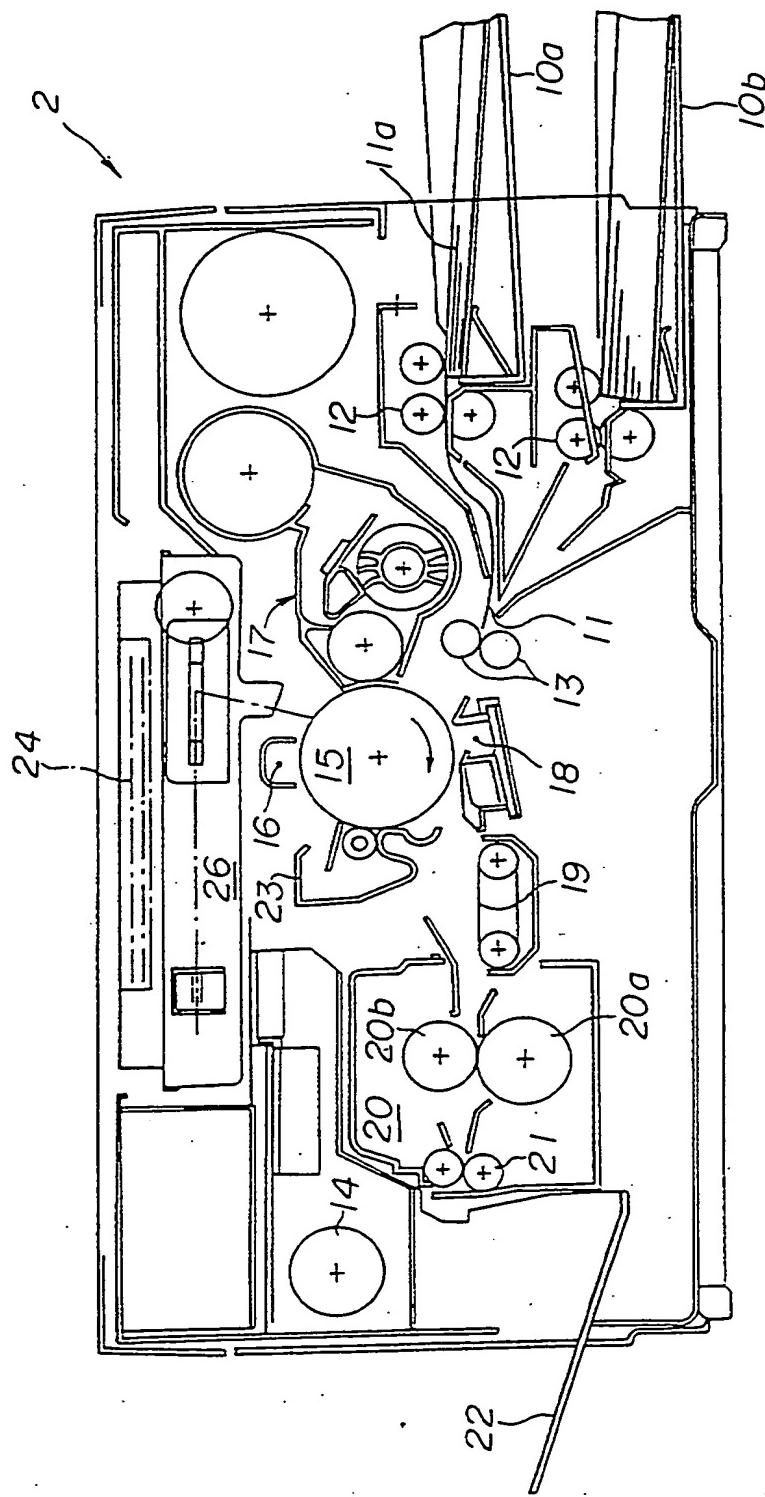


FIG. 3

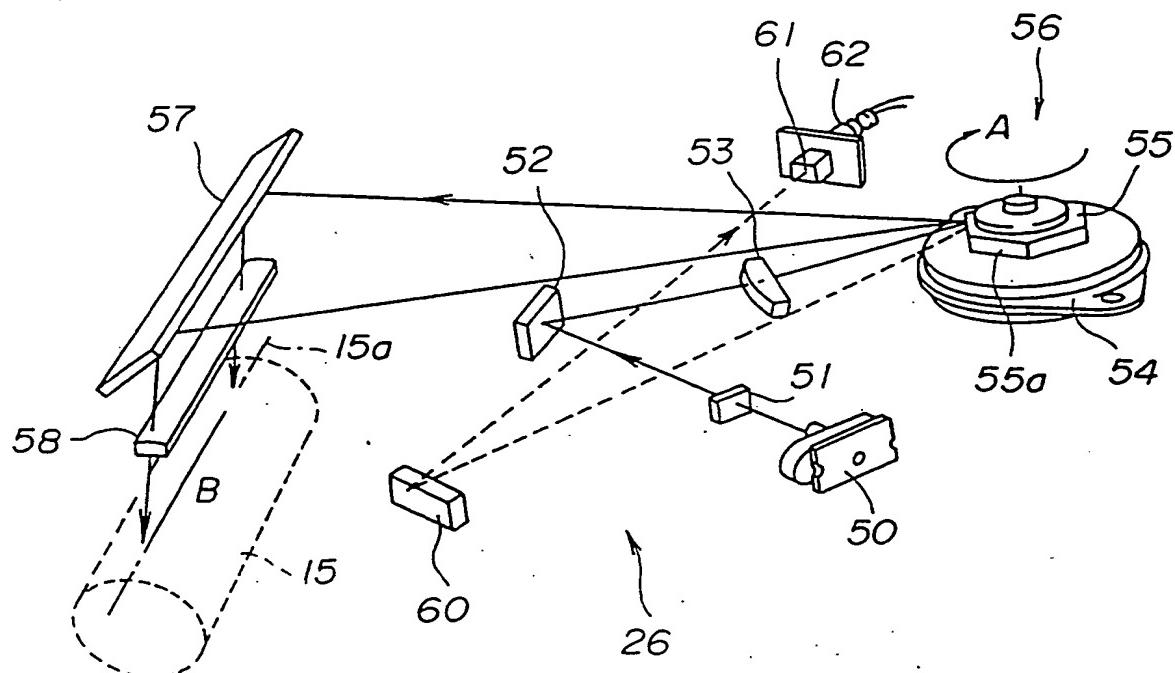


FIG. 4

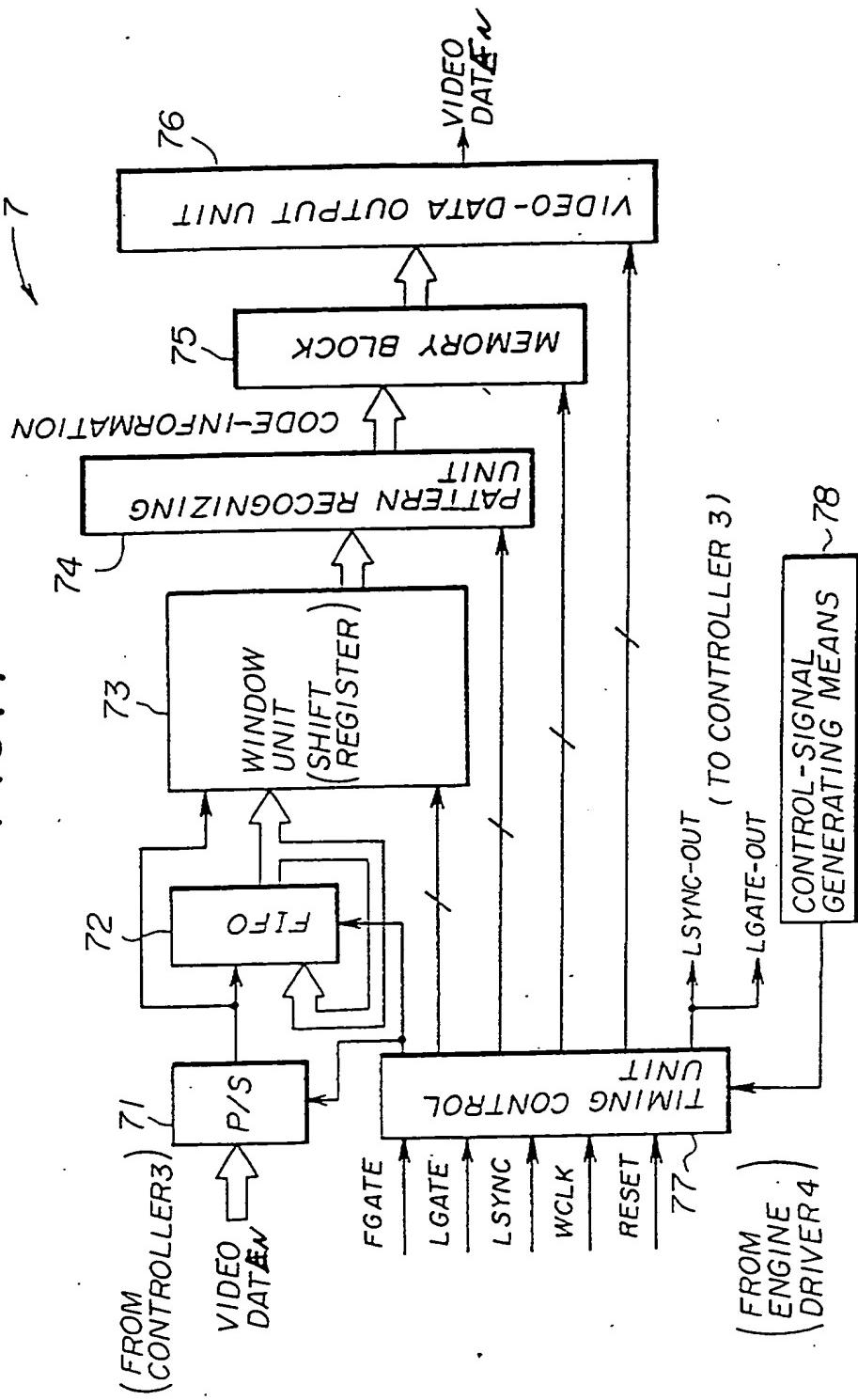
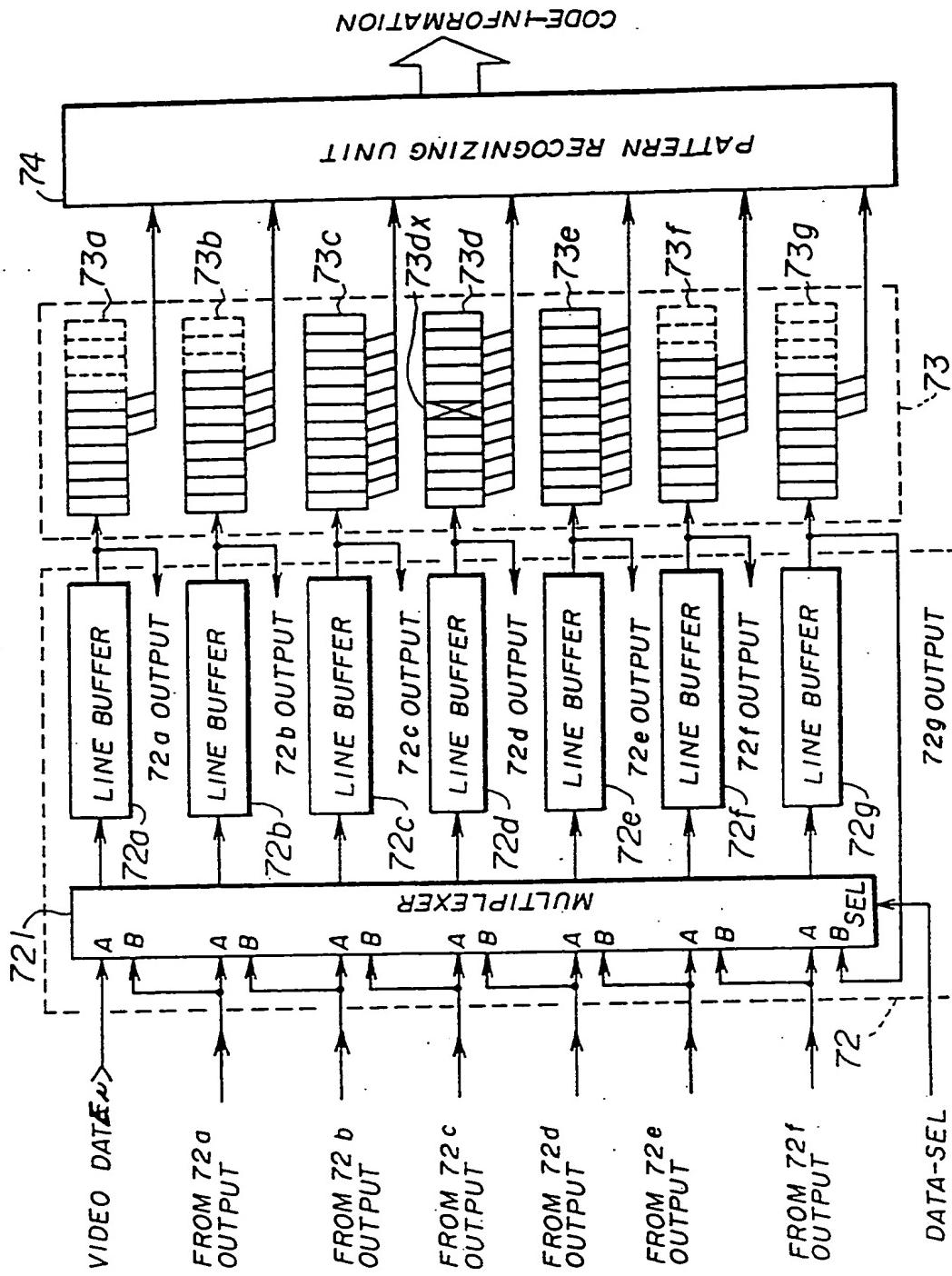
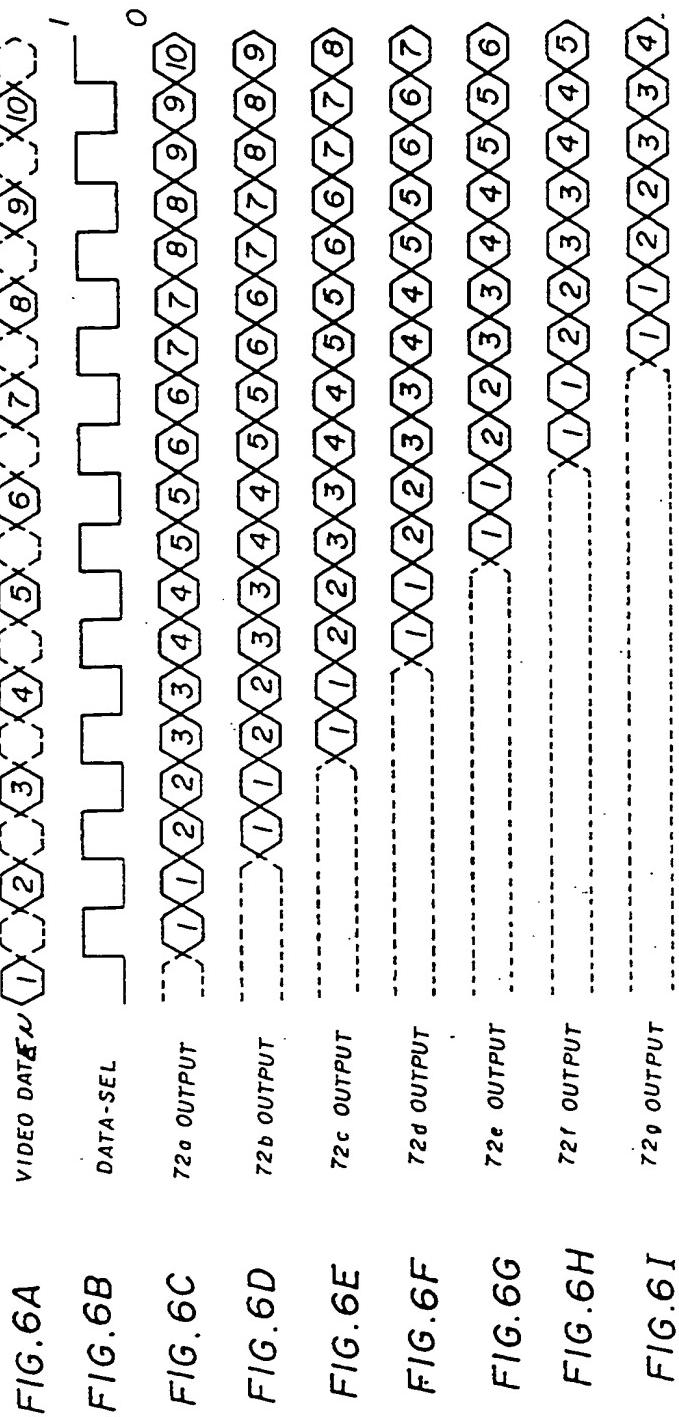
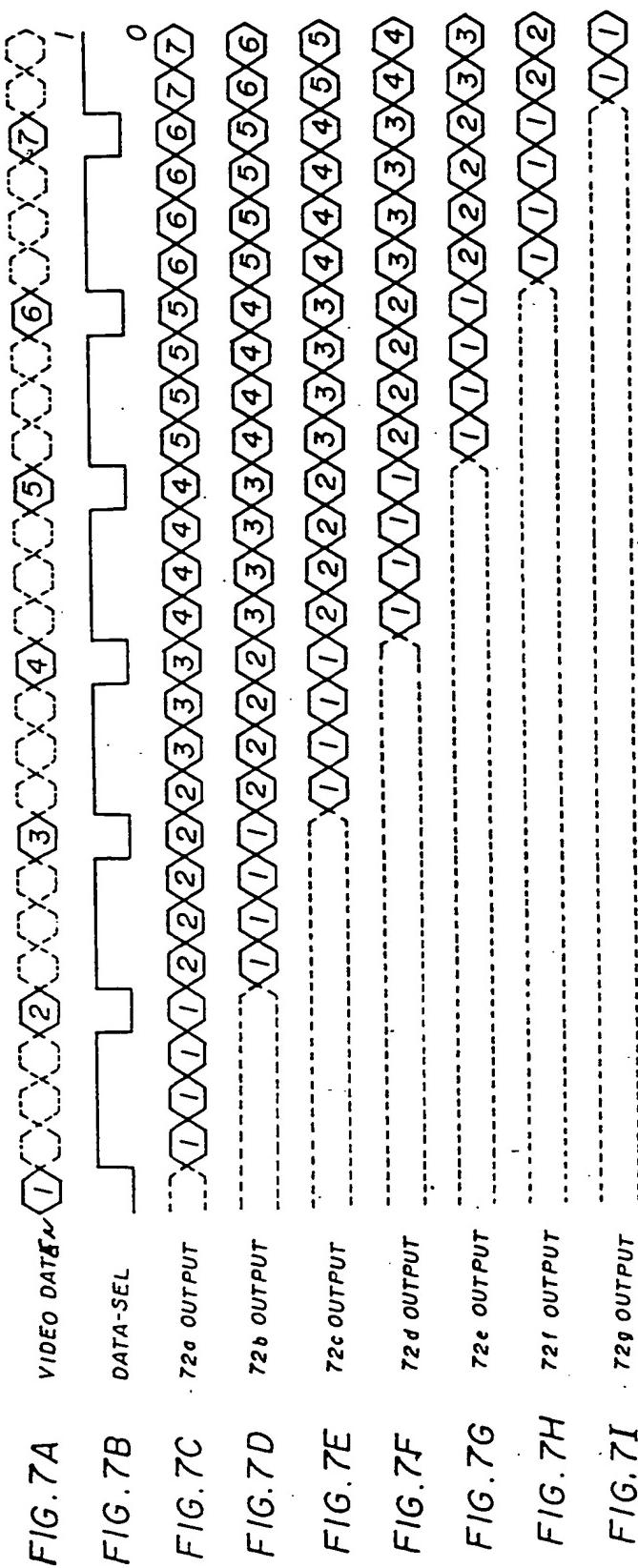


FIG. 5







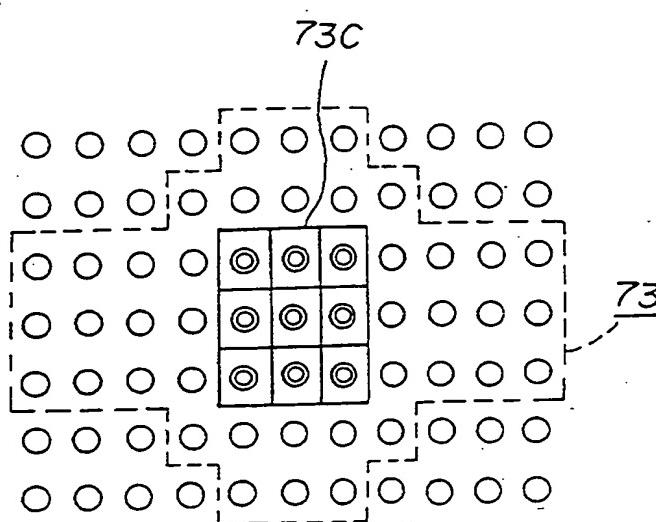
**FIG.8**

FIG. 9

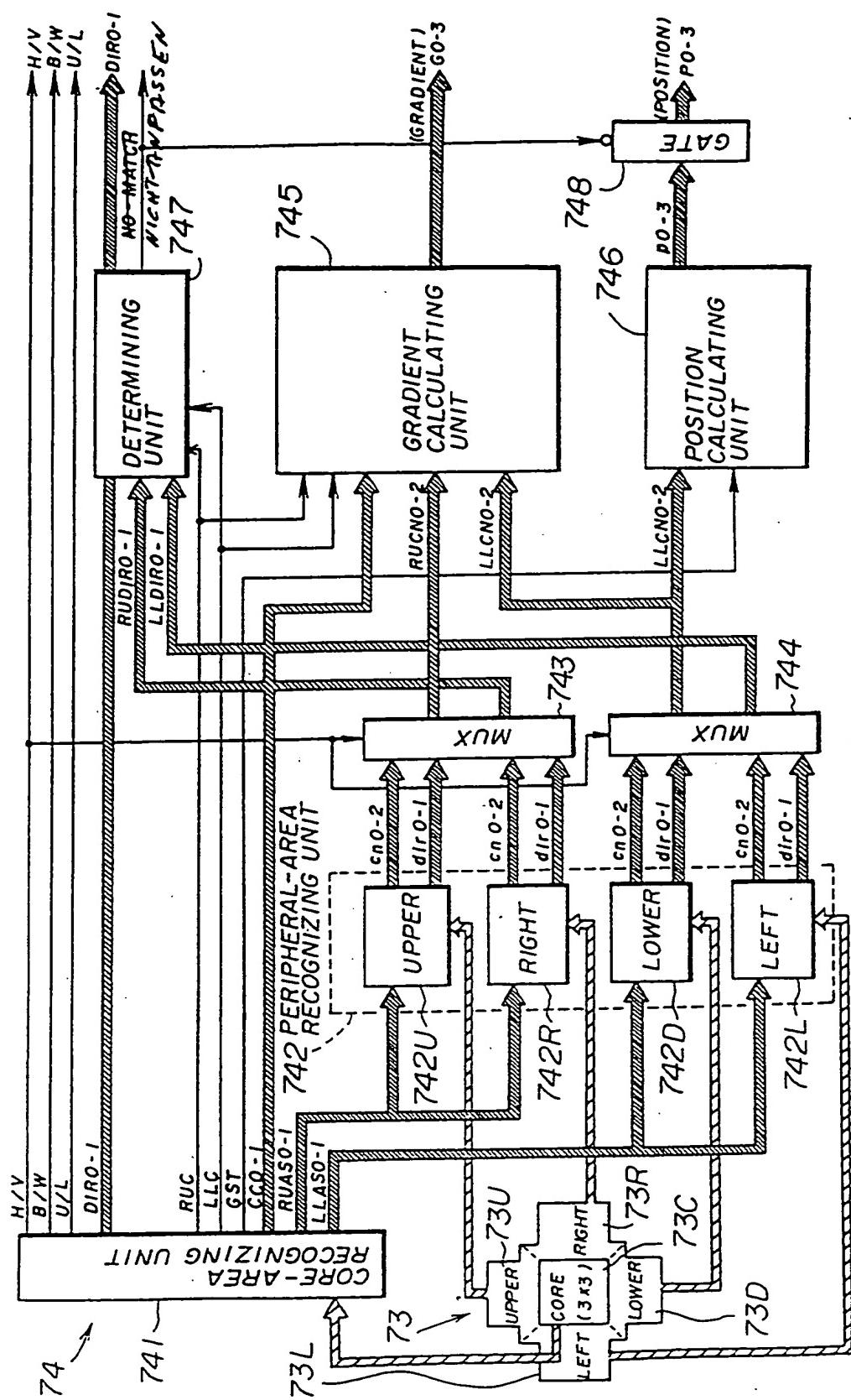


FIG. 10

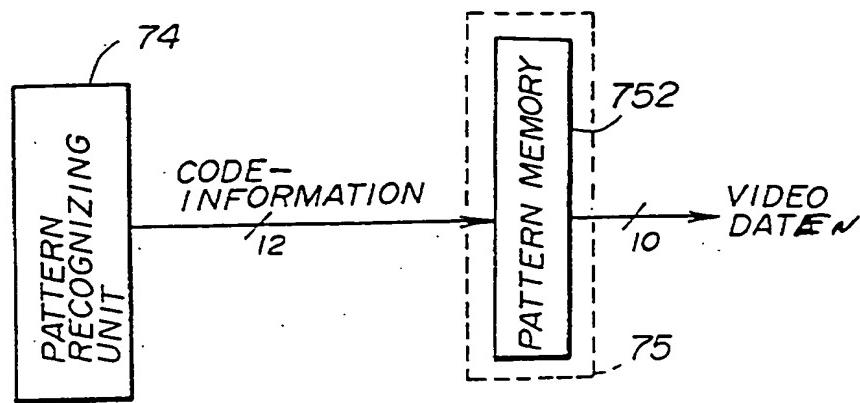


FIG. 11

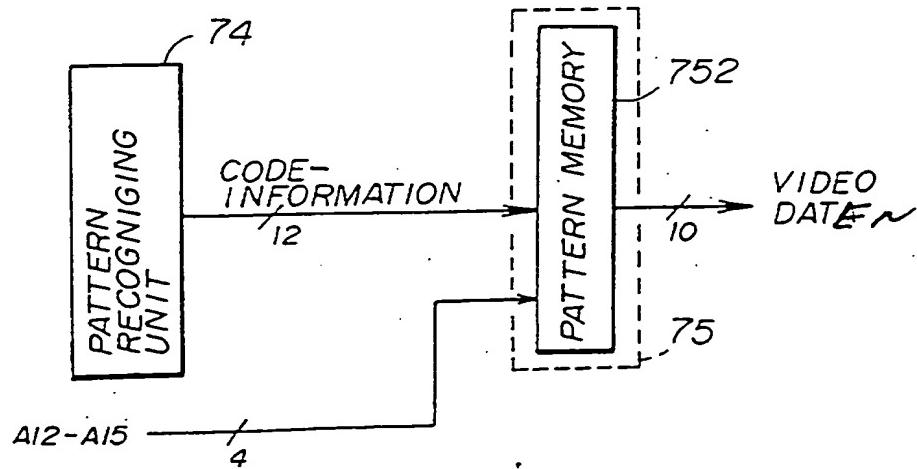


FIG. 12

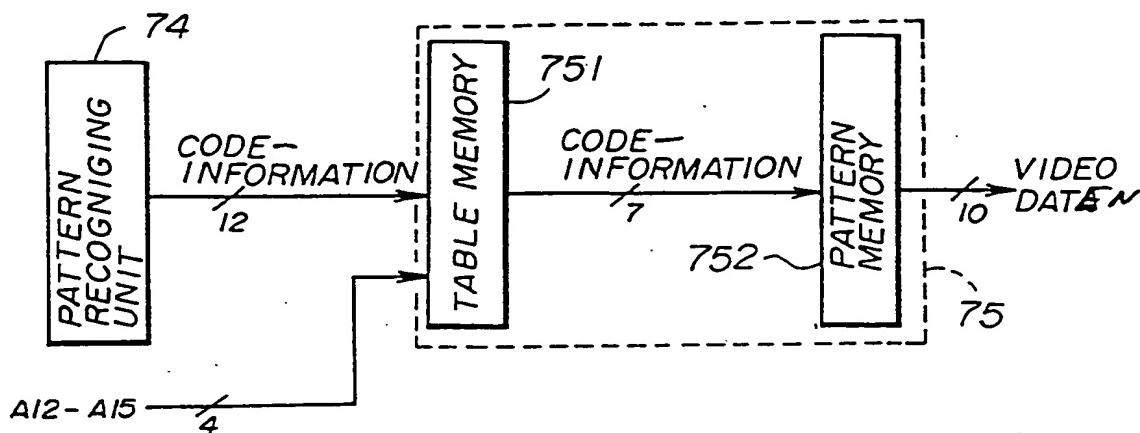
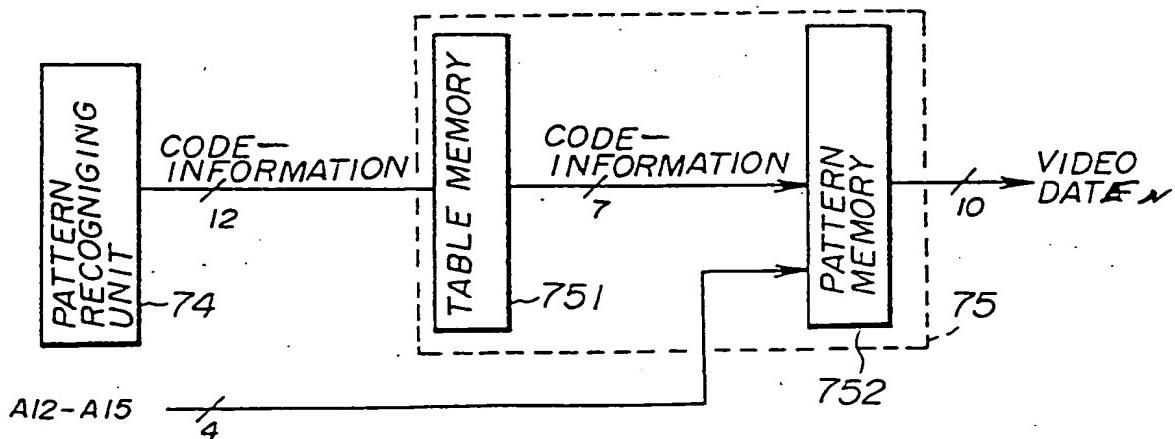


FIG. 13



## FIG. 14

EMBODIMENT	TABLE MEMORY CAPACITY(A)	PATTERN MEMORY CAPACITY(B)	TOTAL CAPACITY(A+B)
FIG.10	0 bit	40960 bit	40960 bit
FIG.11	0 bit	655360 bit	655360 bit
FIG.12	458752 bit	1280 bit	460032 bit
FIG.13	28672 bit	20480 bit	49152 bit

FIG. 15

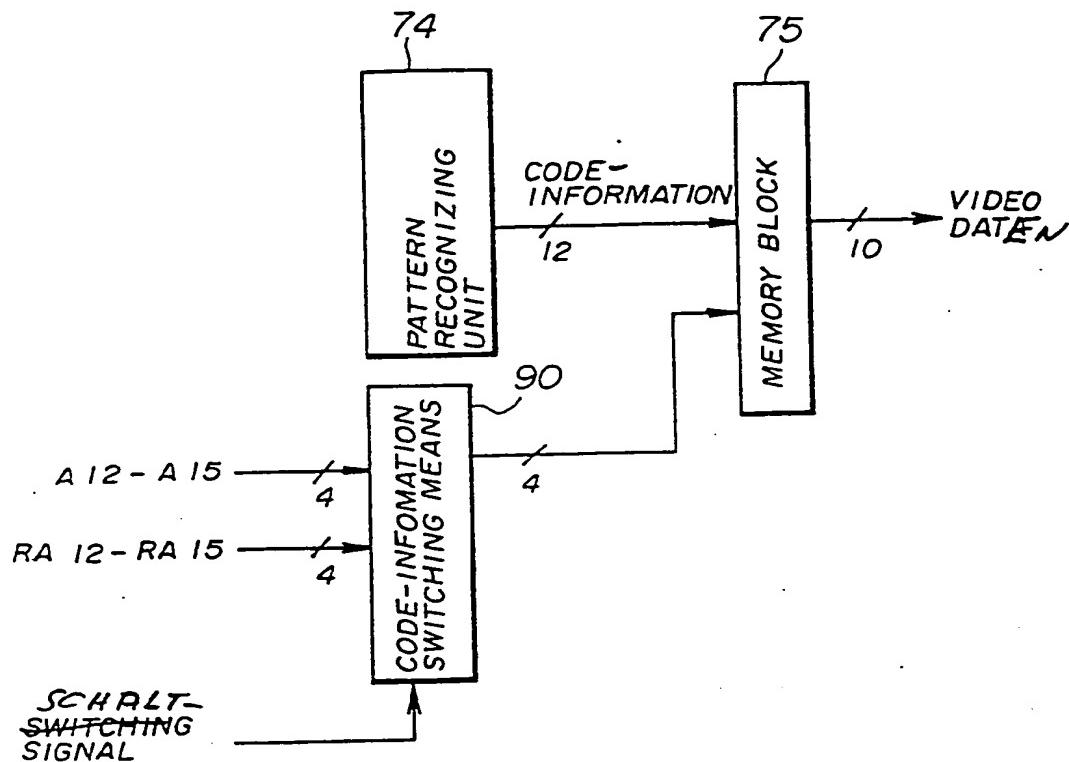
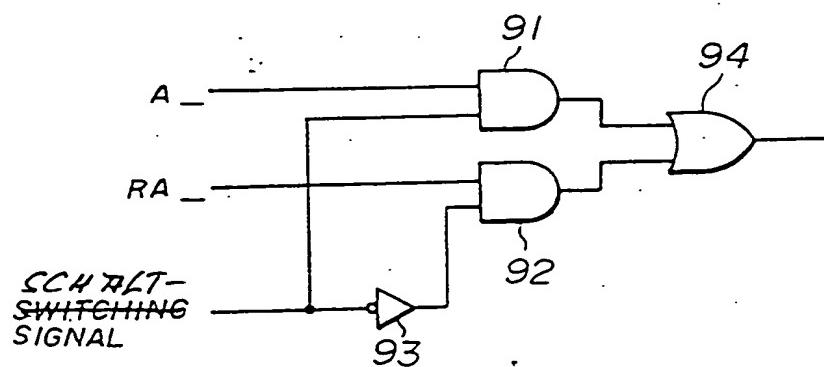


FIG. 16



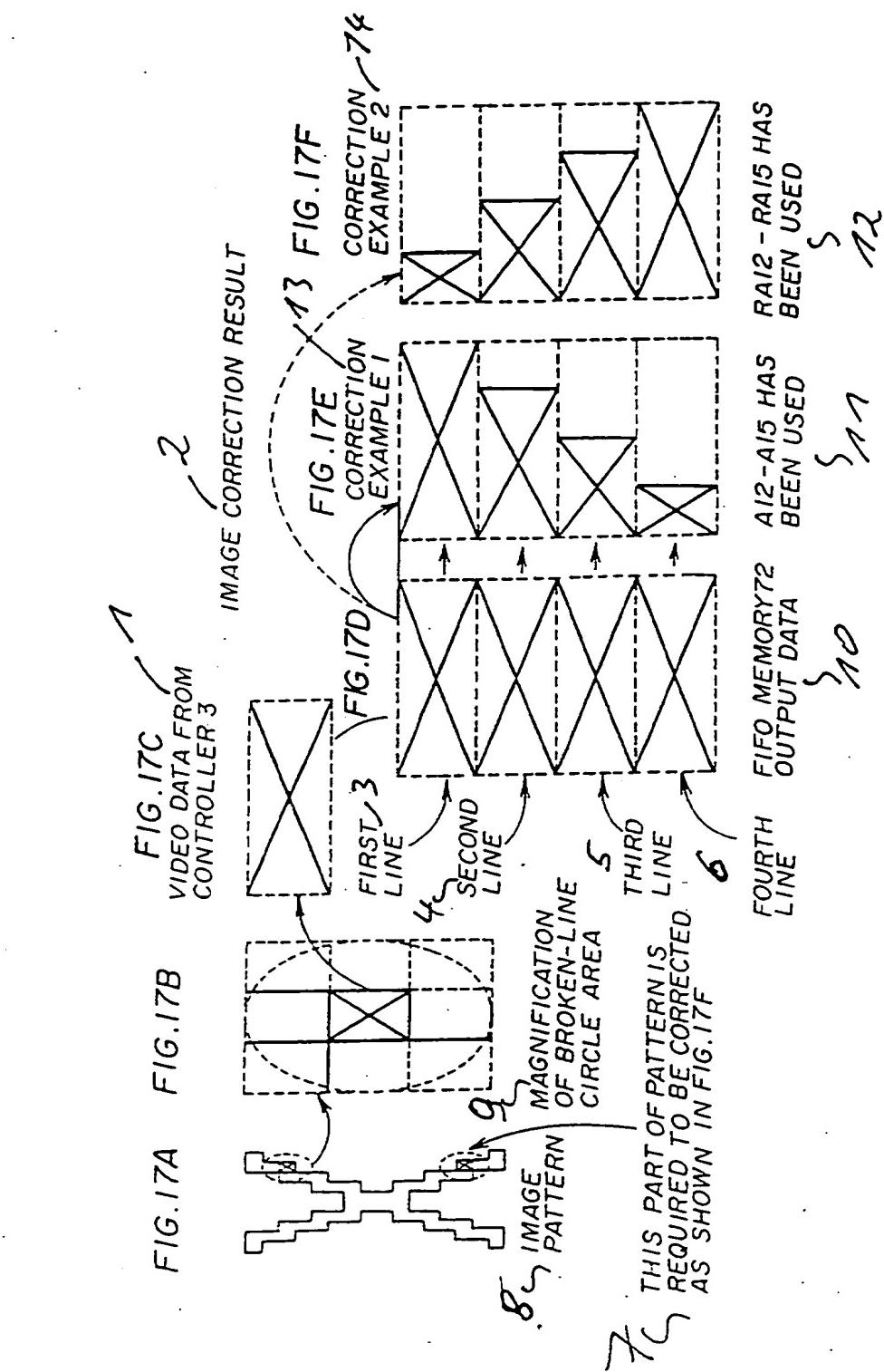


FIG. 18

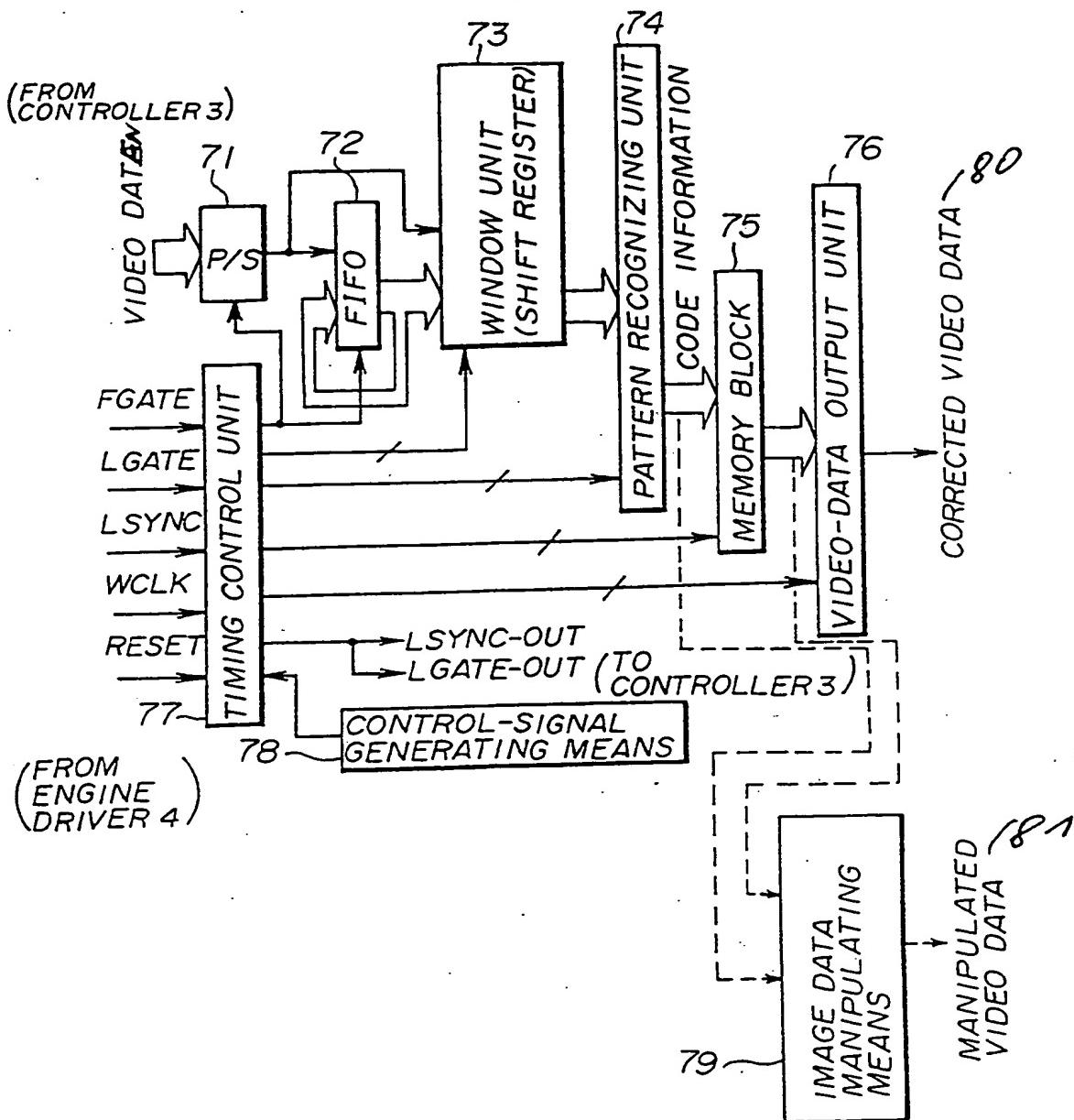


FIG. 19

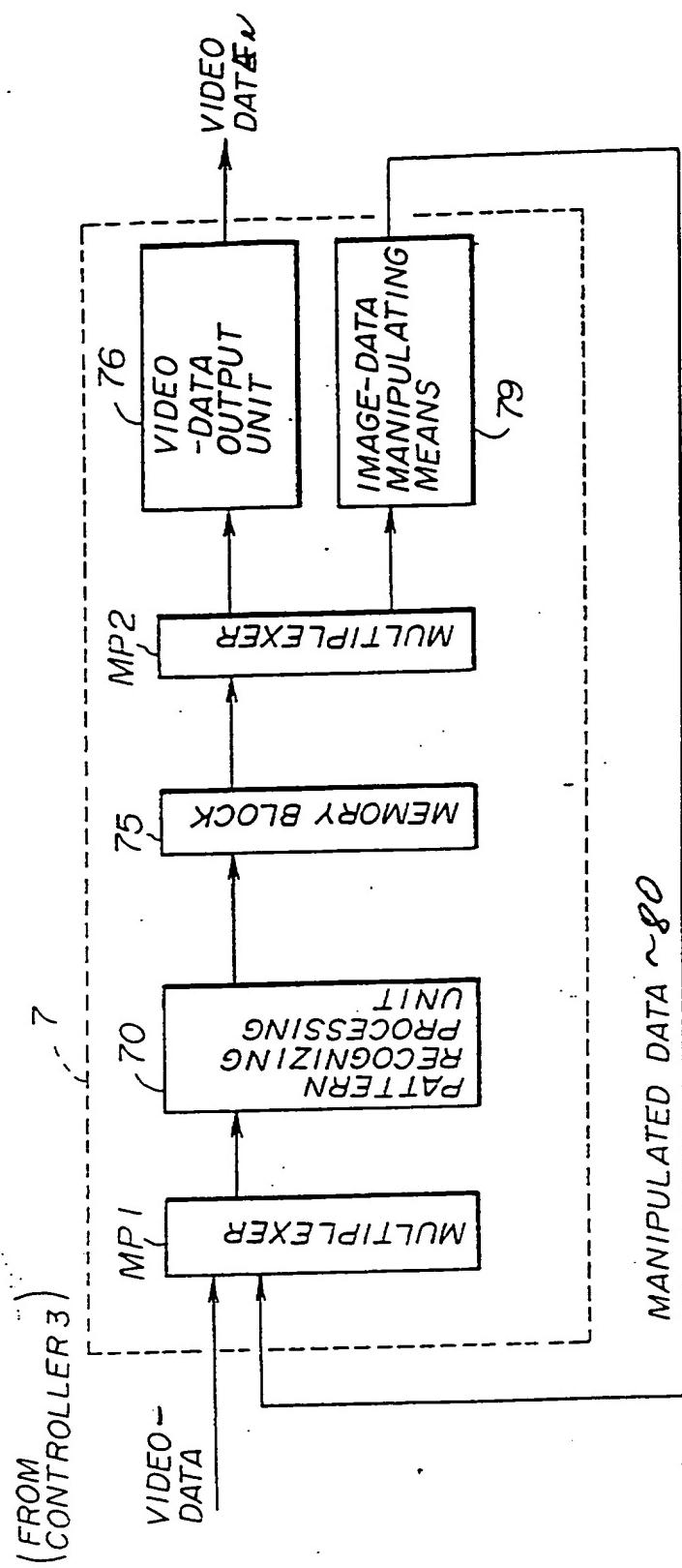


FIG. 20

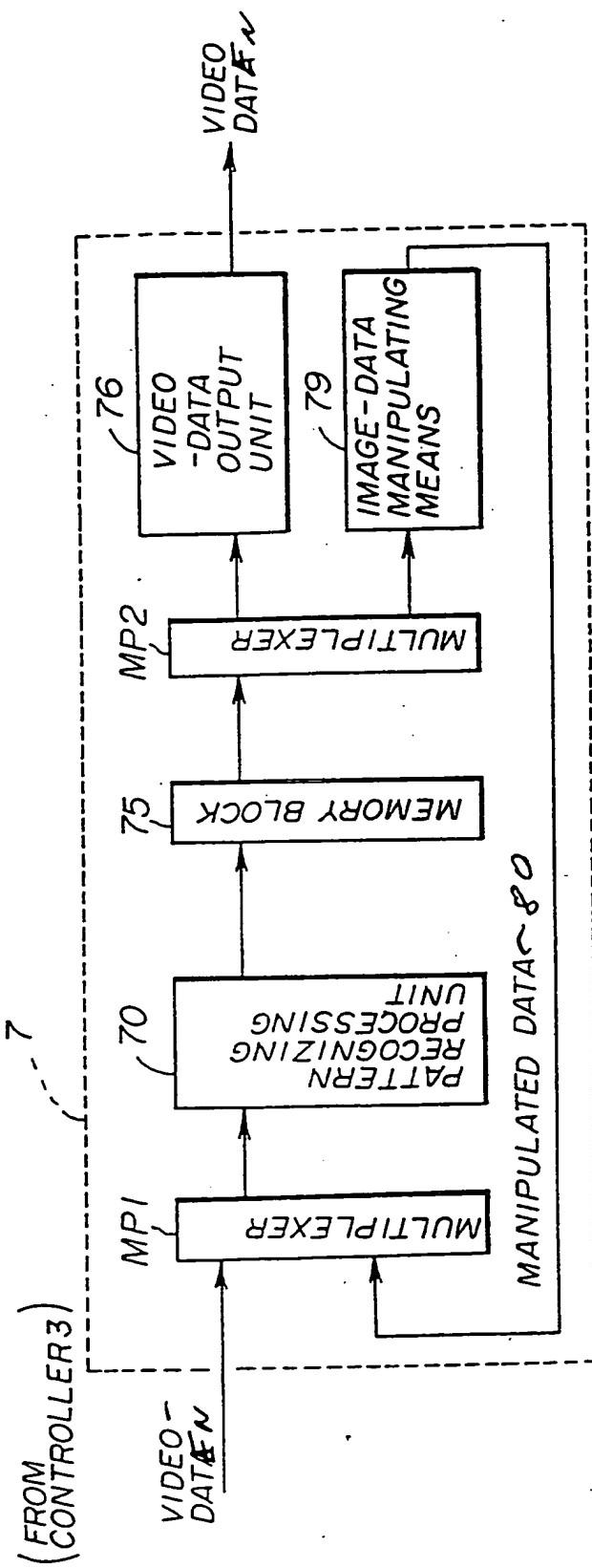
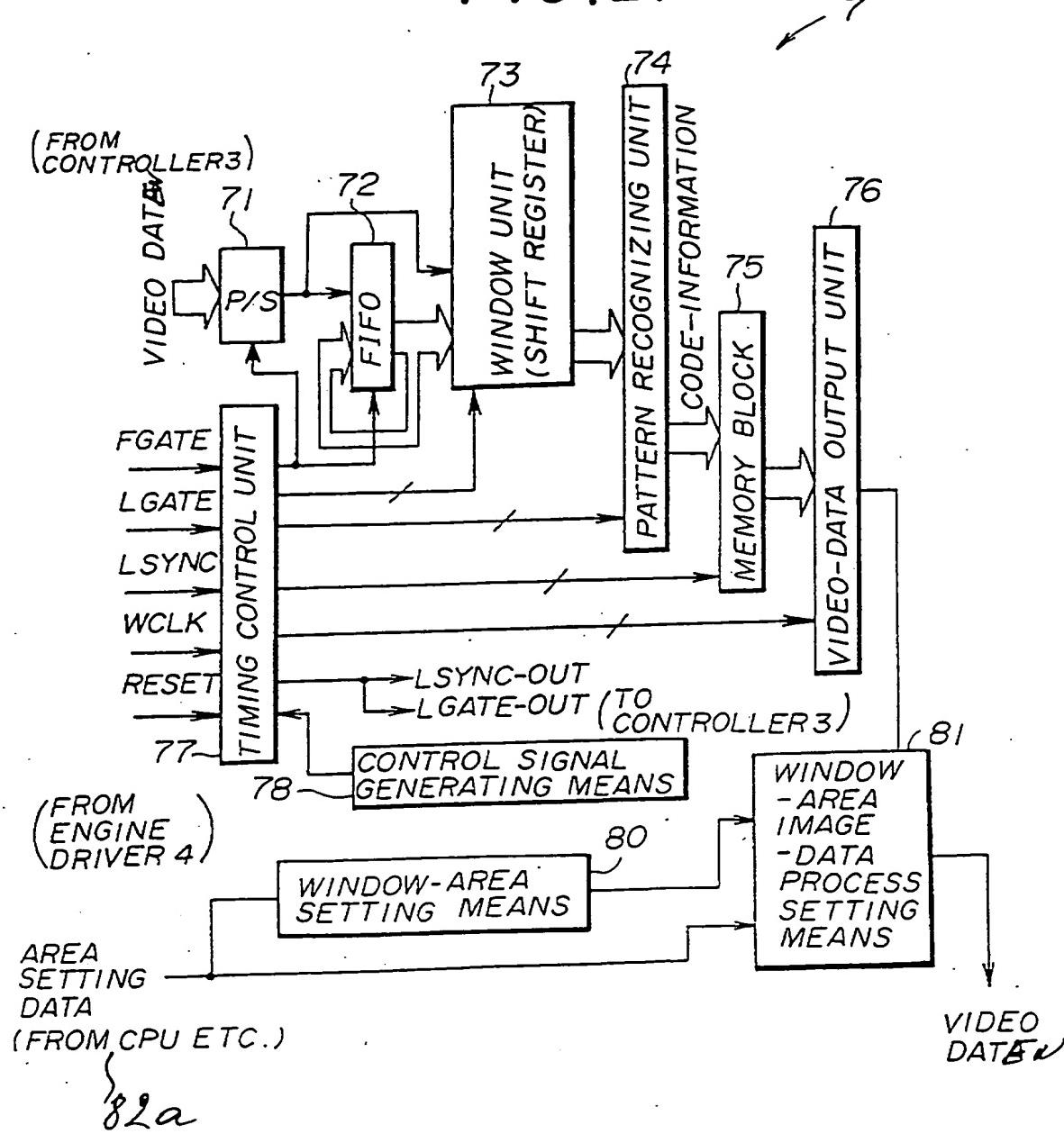


FIG. 21



## FIG. 22

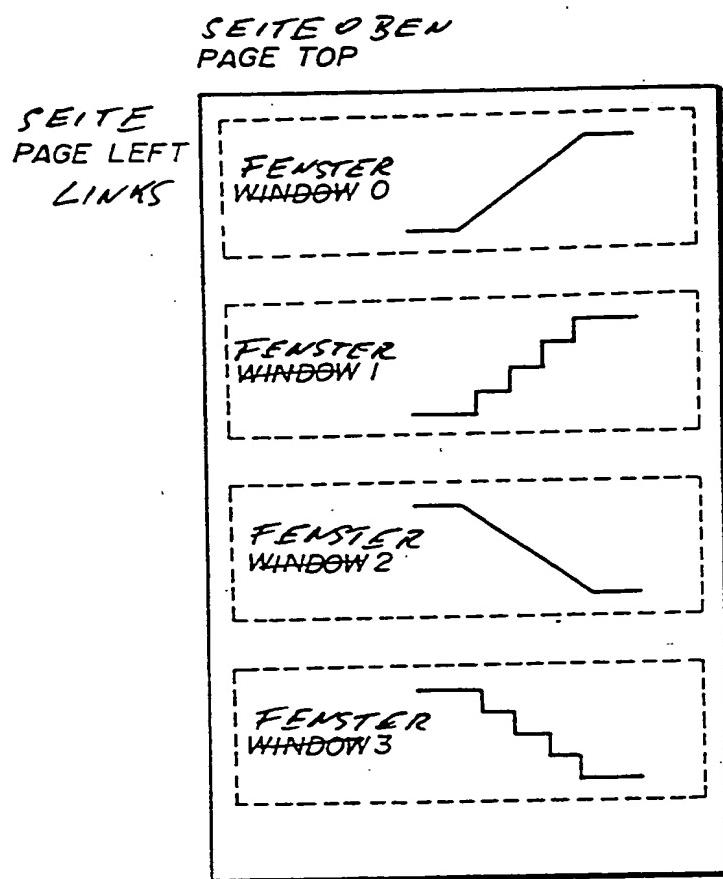
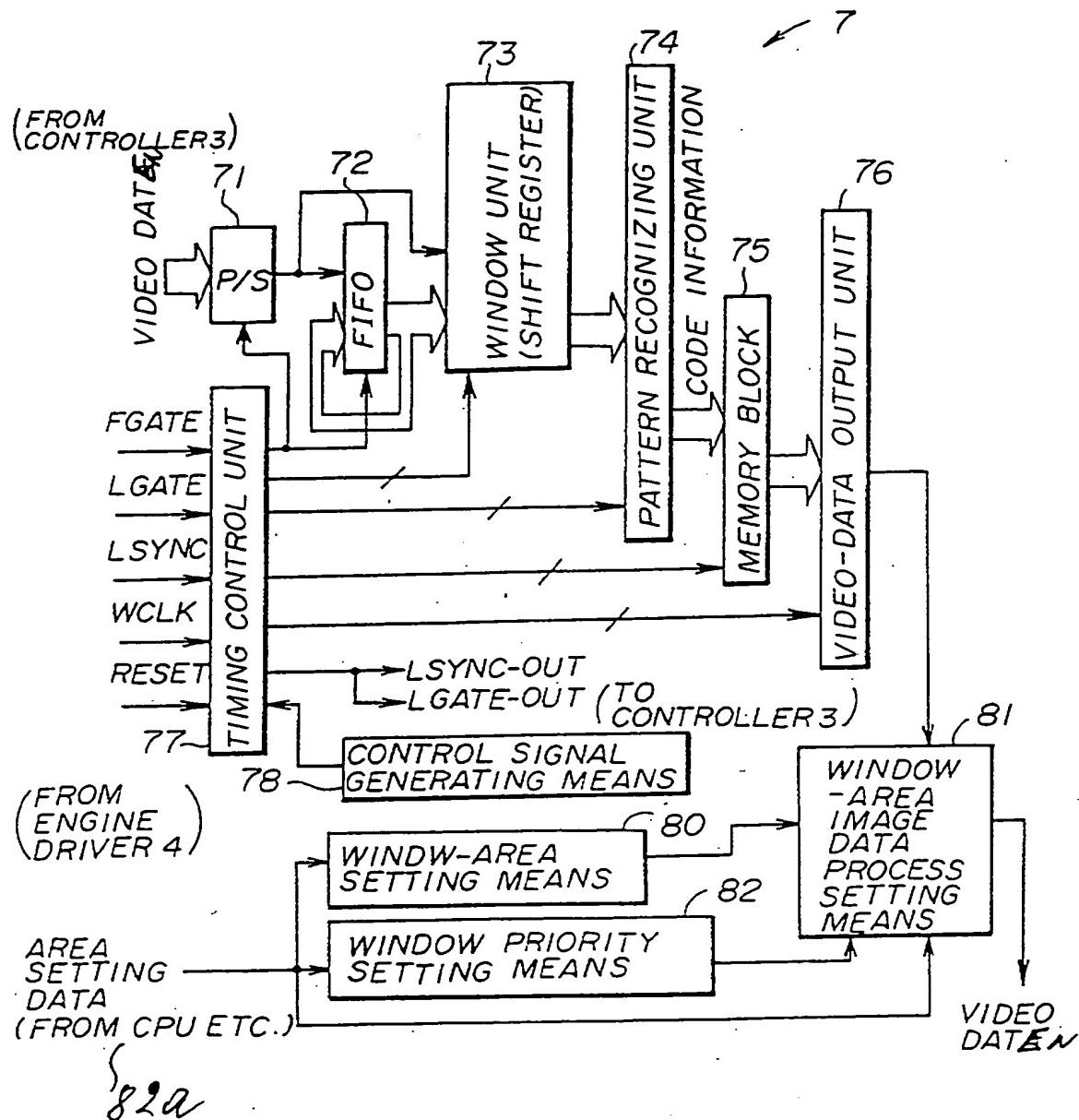


FIG. 23



## FIG. 24

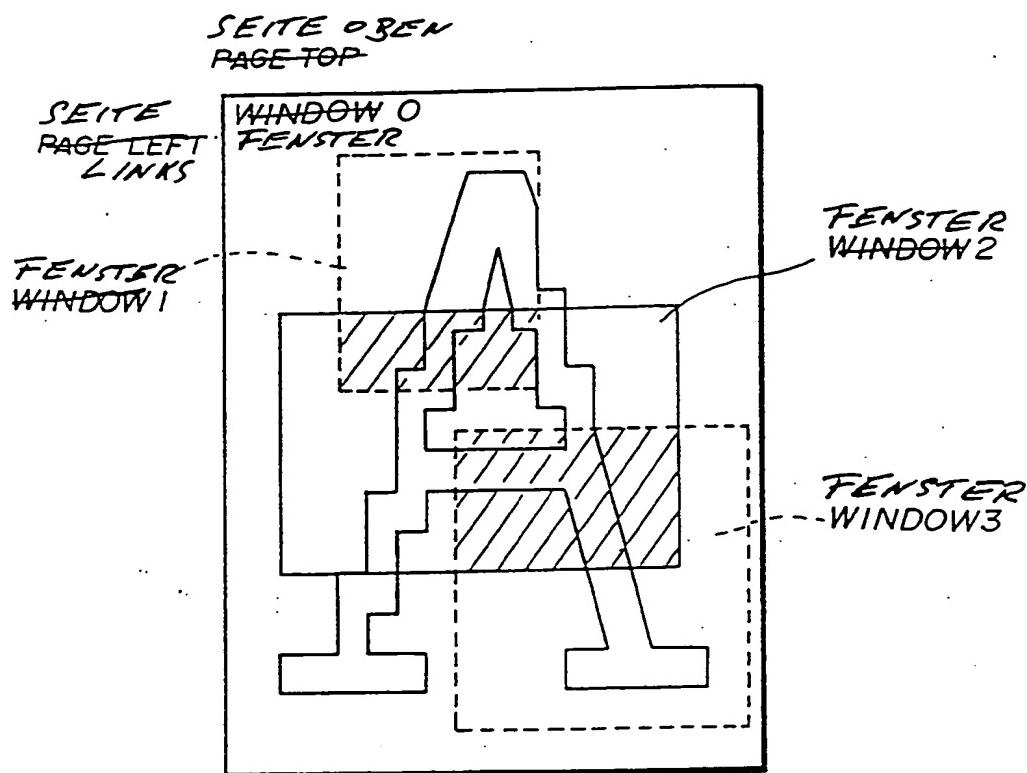
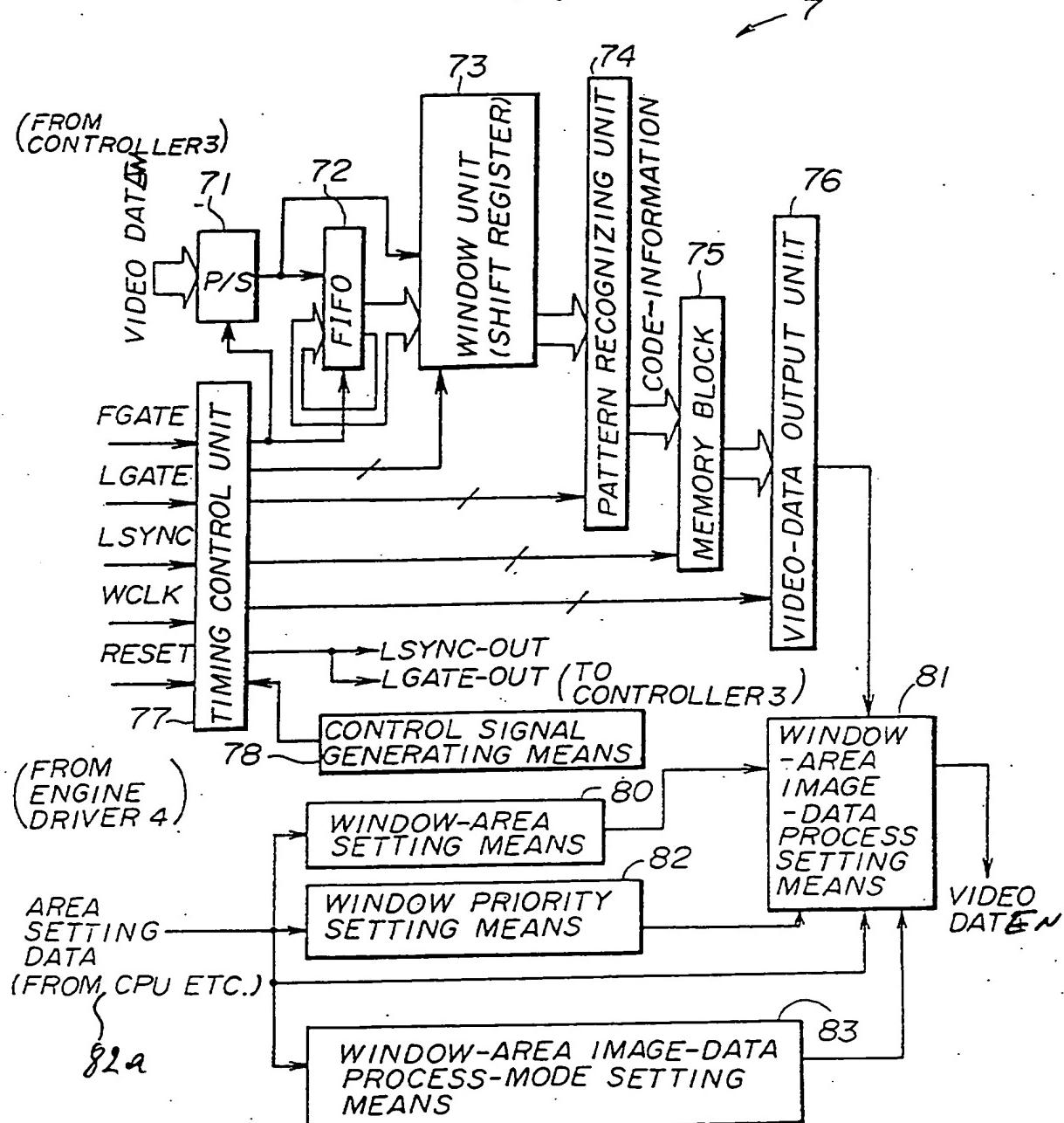


FIG.25



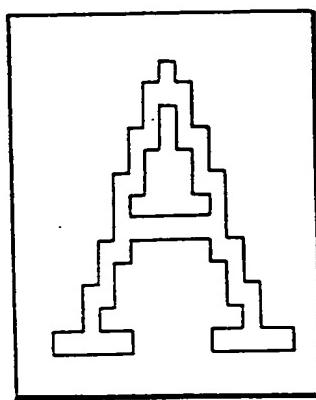
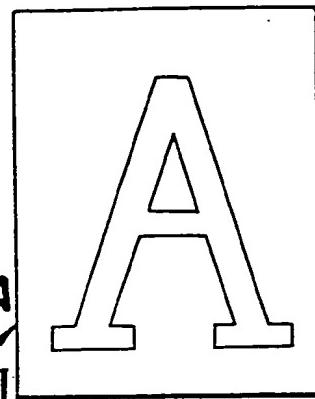
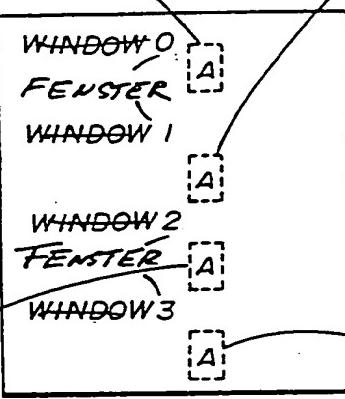
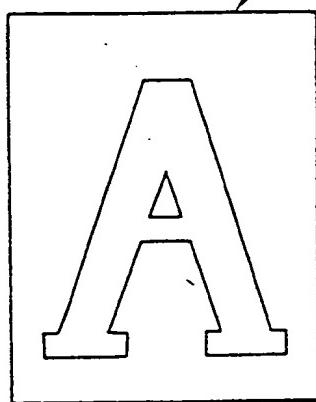
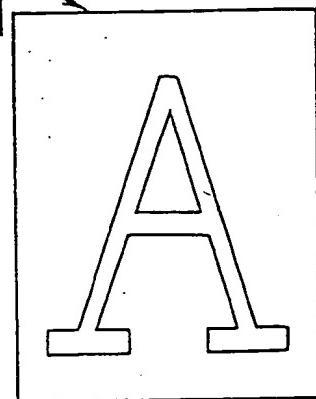
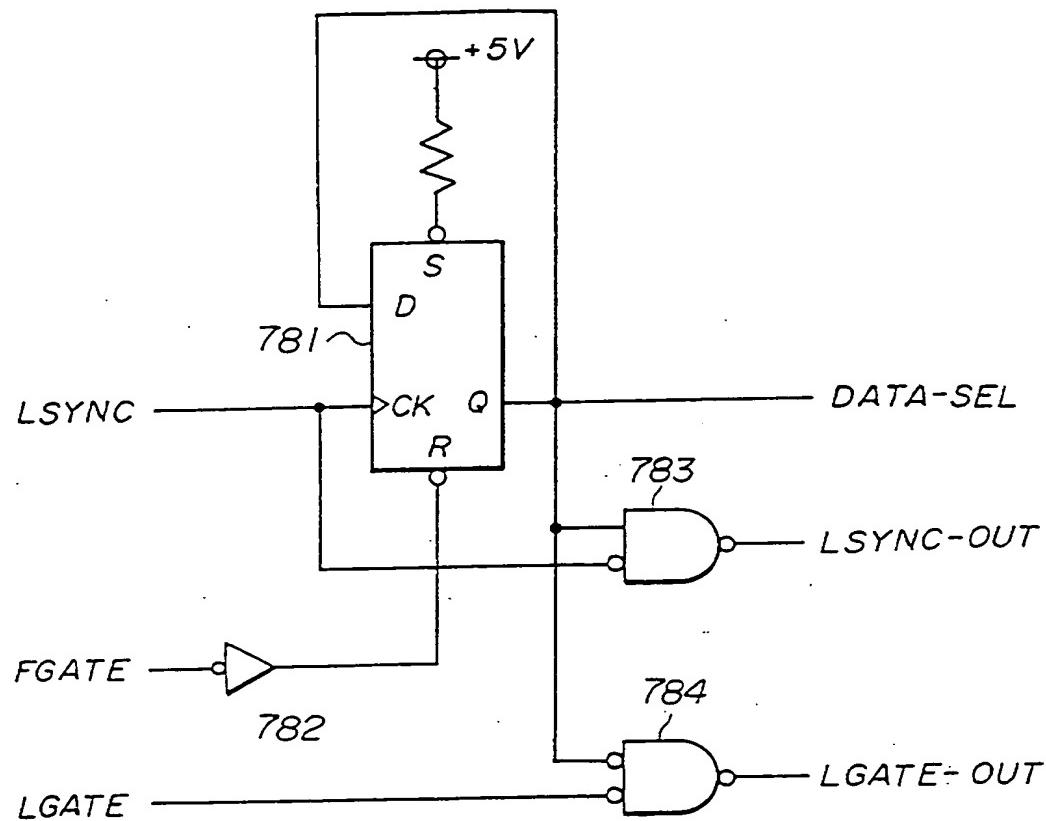
**FIG. 26B****FIG. 26C****FIG. 26A****FIG. 26D****FIG. 26E**

FIG. 27



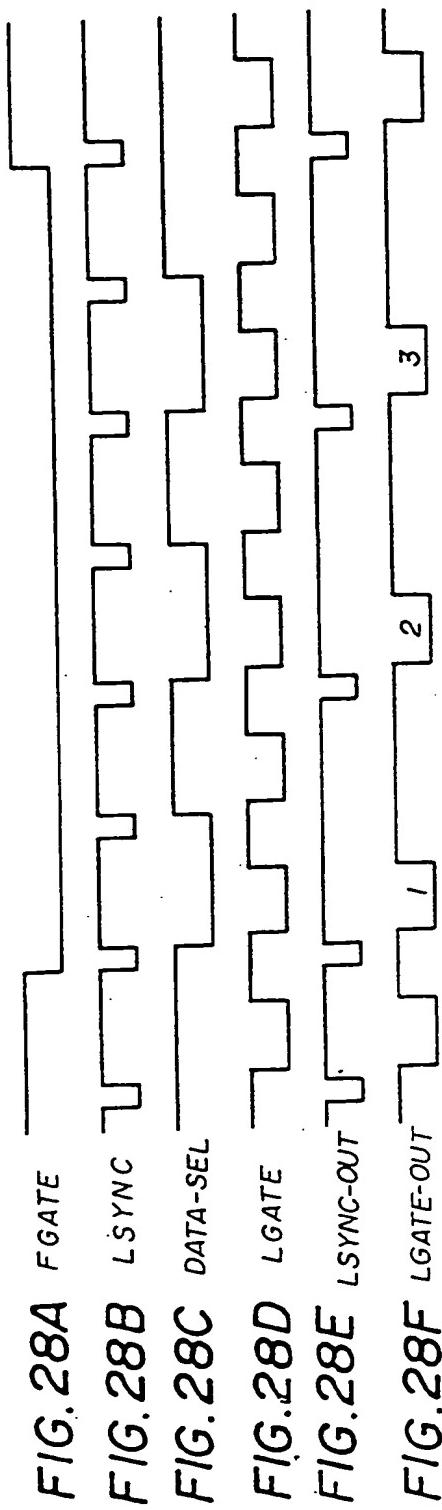
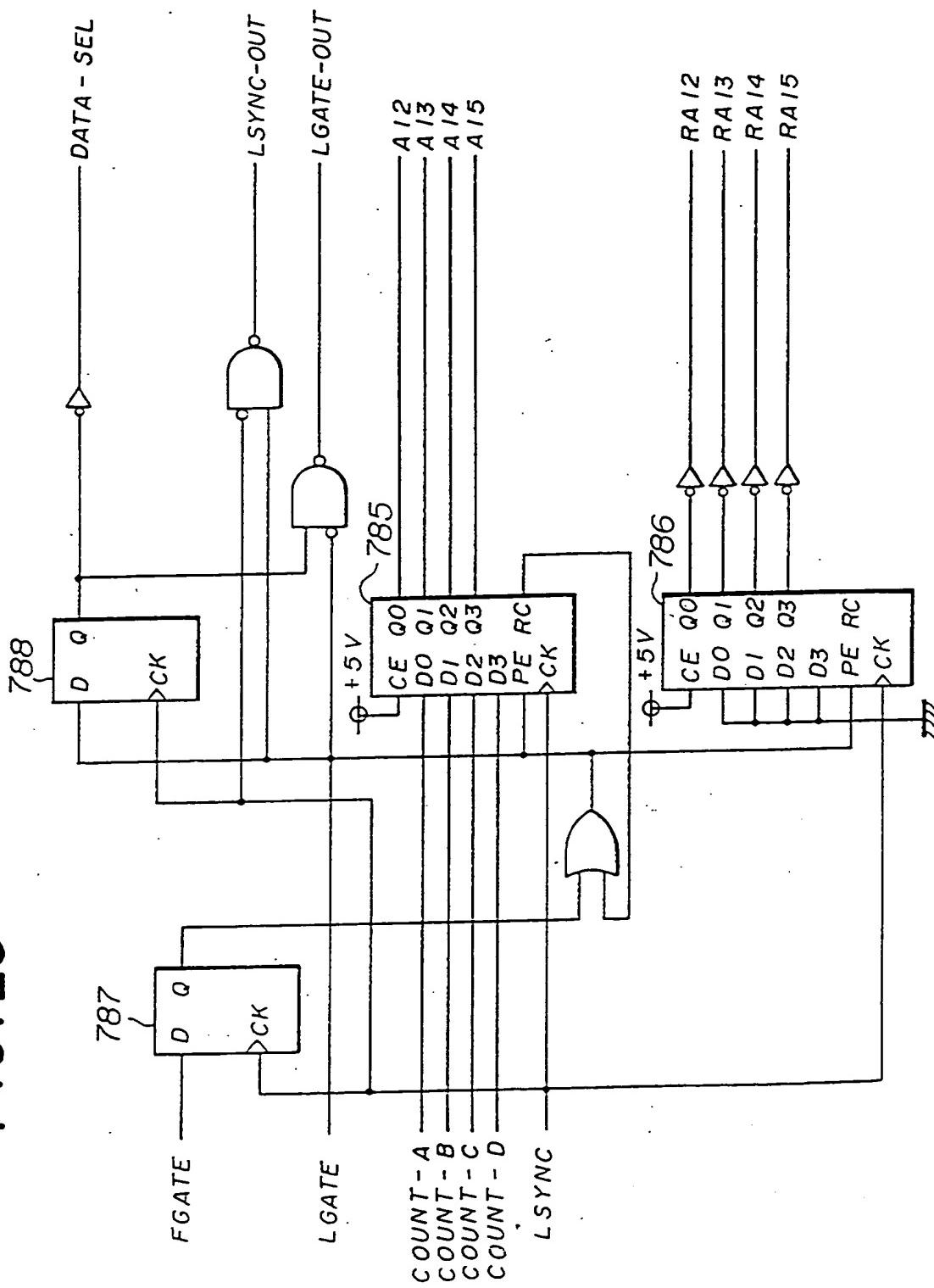


FIG. 29



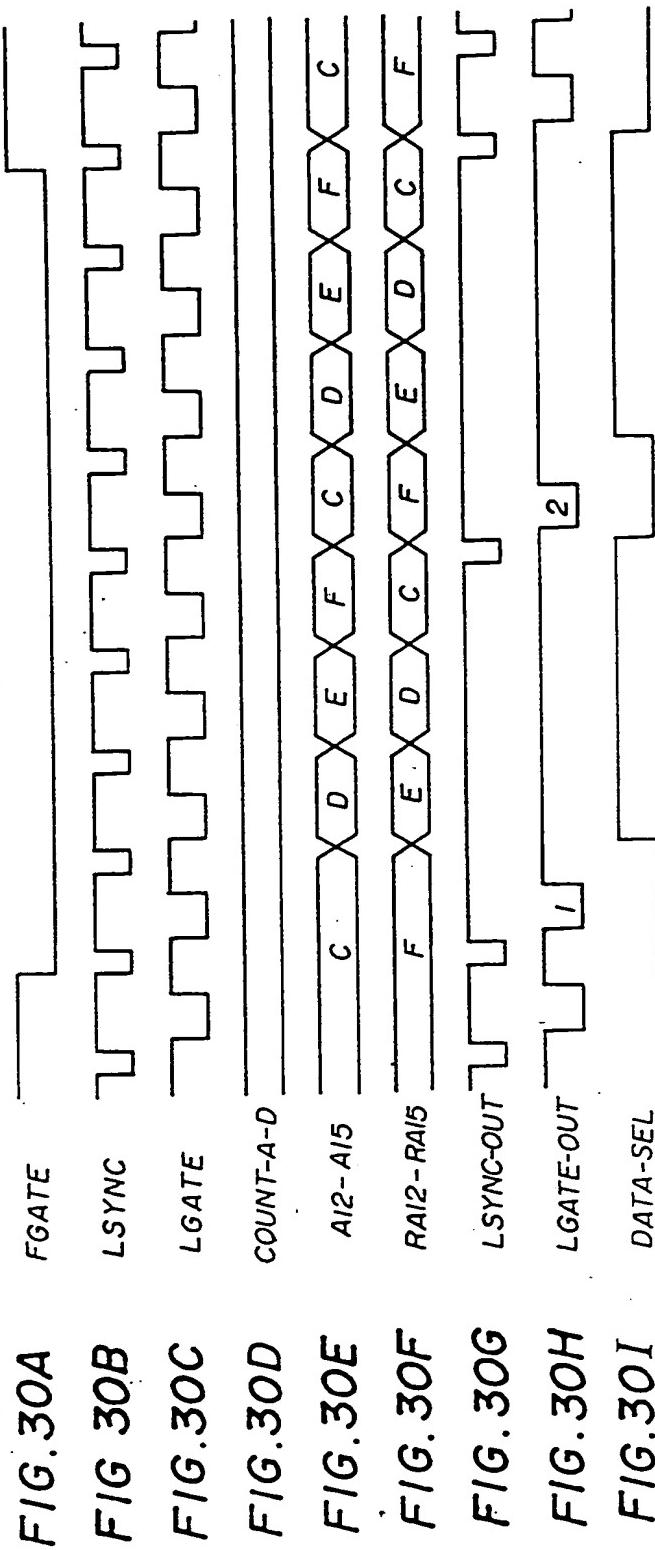


FIG.31A

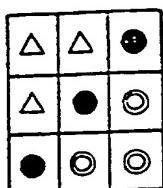


FIG.31B

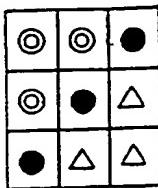


FIG.31C

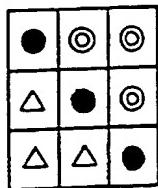


FIG.31D

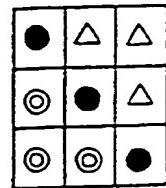


FIG.32A

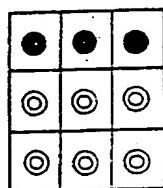


FIG.32B

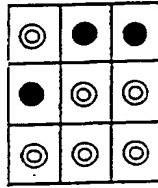


FIG.32C

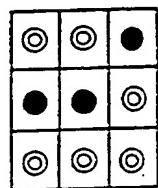


FIG.32D

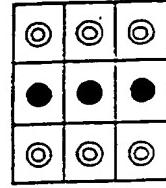


FIG.32E

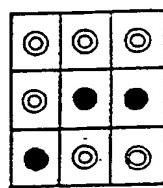


FIG.32F

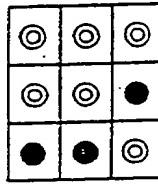


FIG.32G

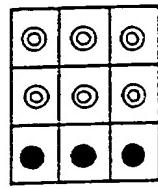


FIG.33A

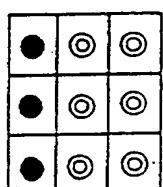


FIG.33B

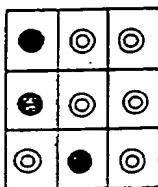


FIG.33C

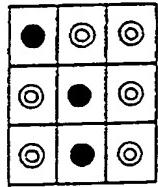


FIG.33D

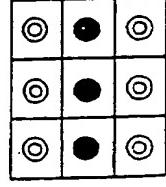


FIG.33E

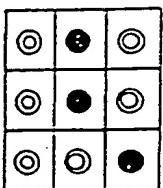


FIG.33F

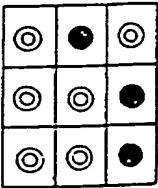
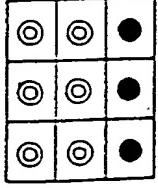
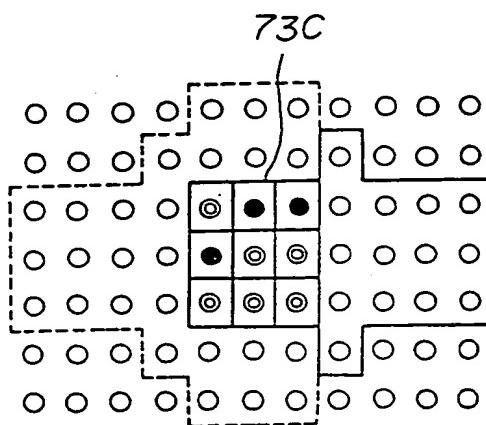
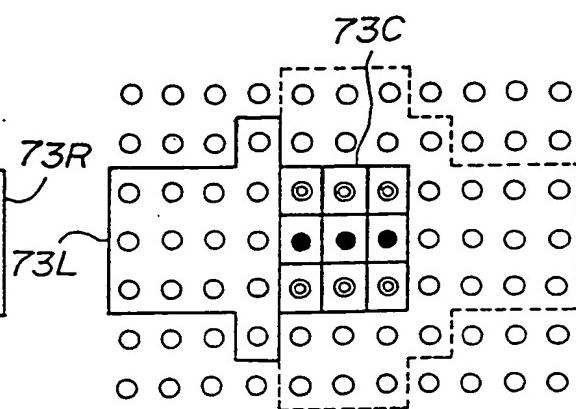
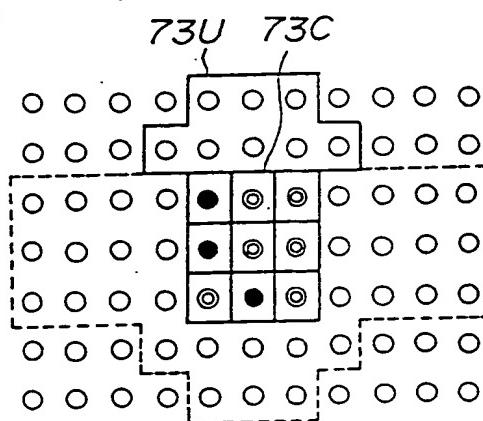
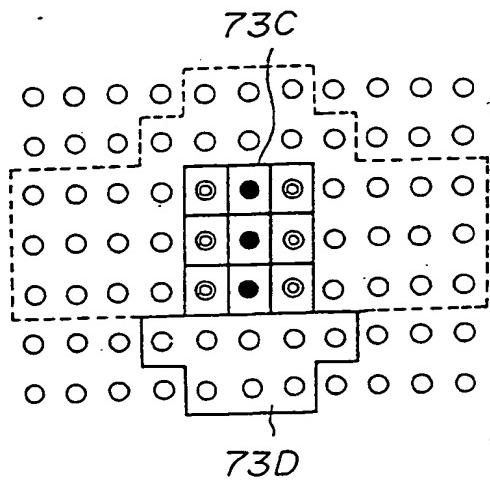
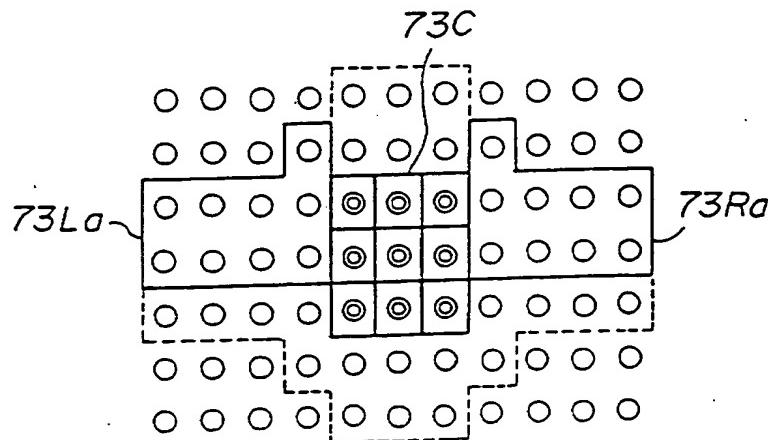
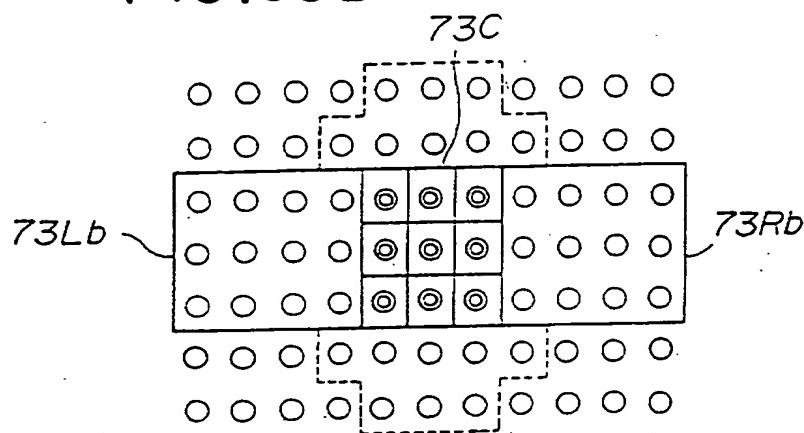
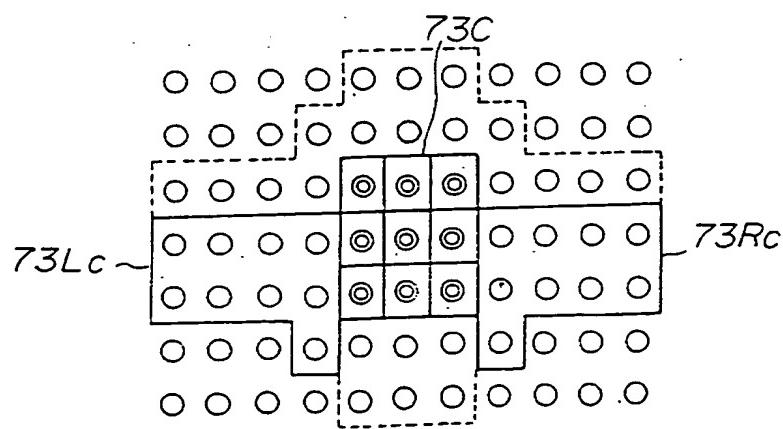
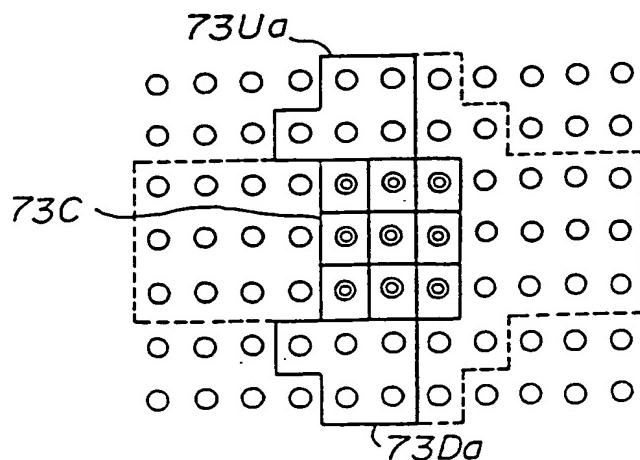
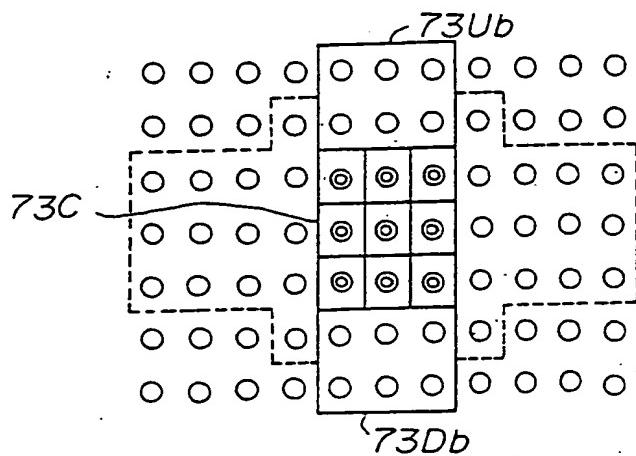
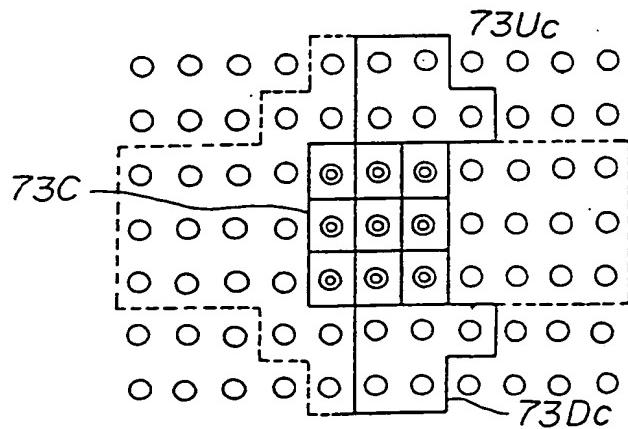


FIG.33G



**FIG. 34A****FIG. 34B****FIG. 34C****FIG. 34D**

**FIG. 35A****FIG. 35B****FIG. 35C**

**FIG. 36A****FIG. 36B****FIG. 36C**

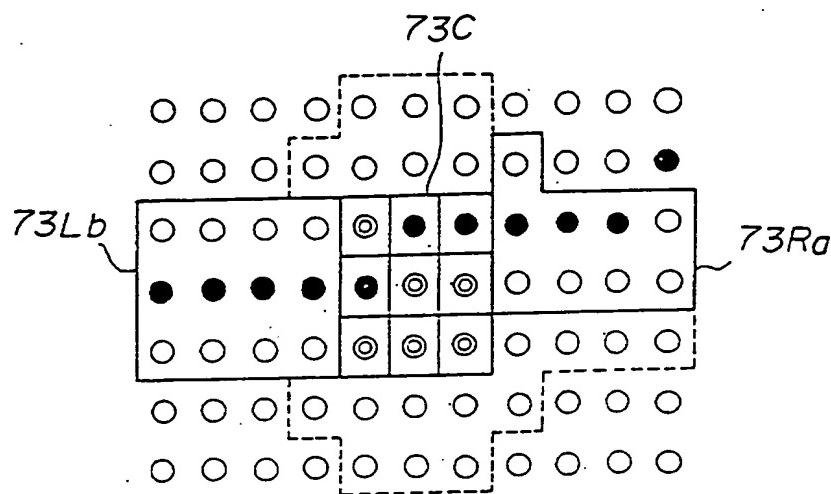
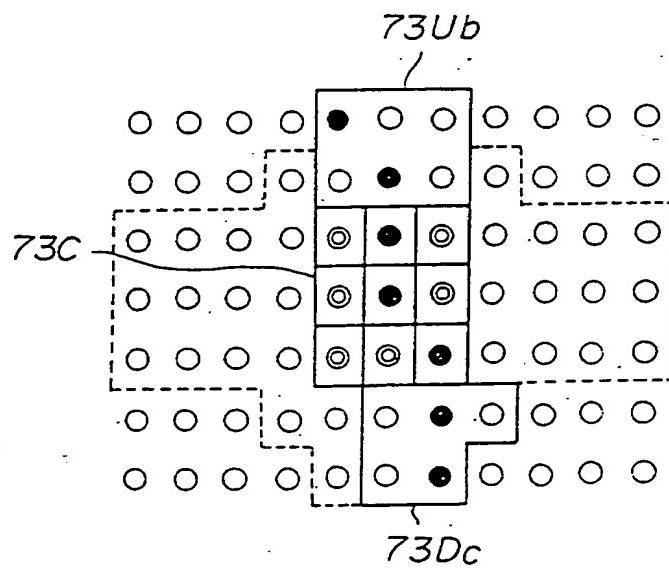
*FIG. 37**FIG. 38*

FIG. 39

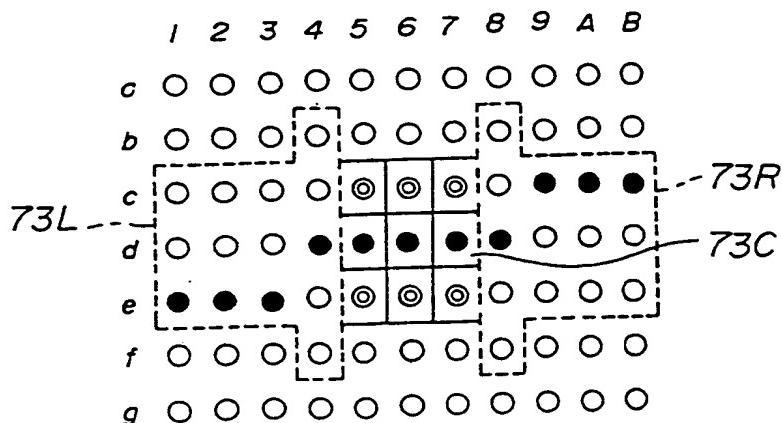


FIG. 40

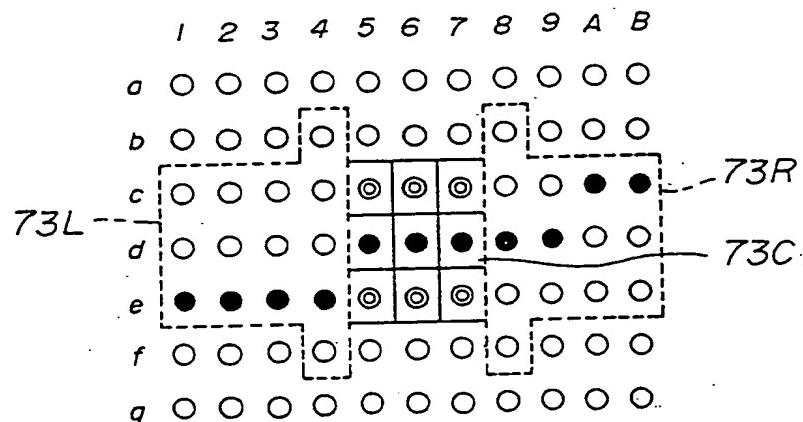


FIG. 41

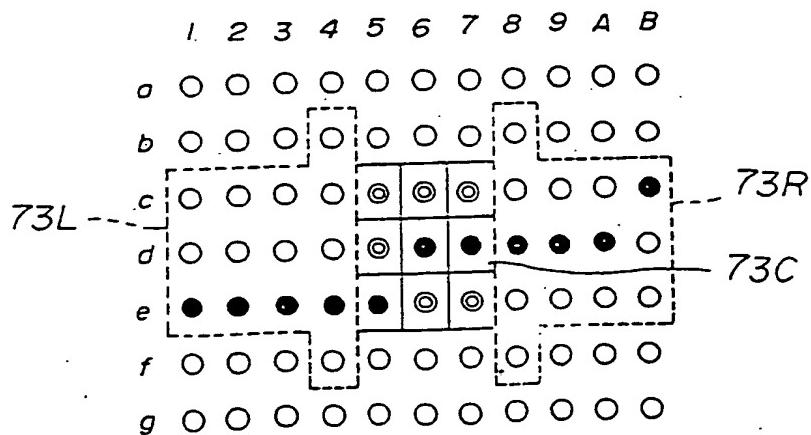


FIG. 42A

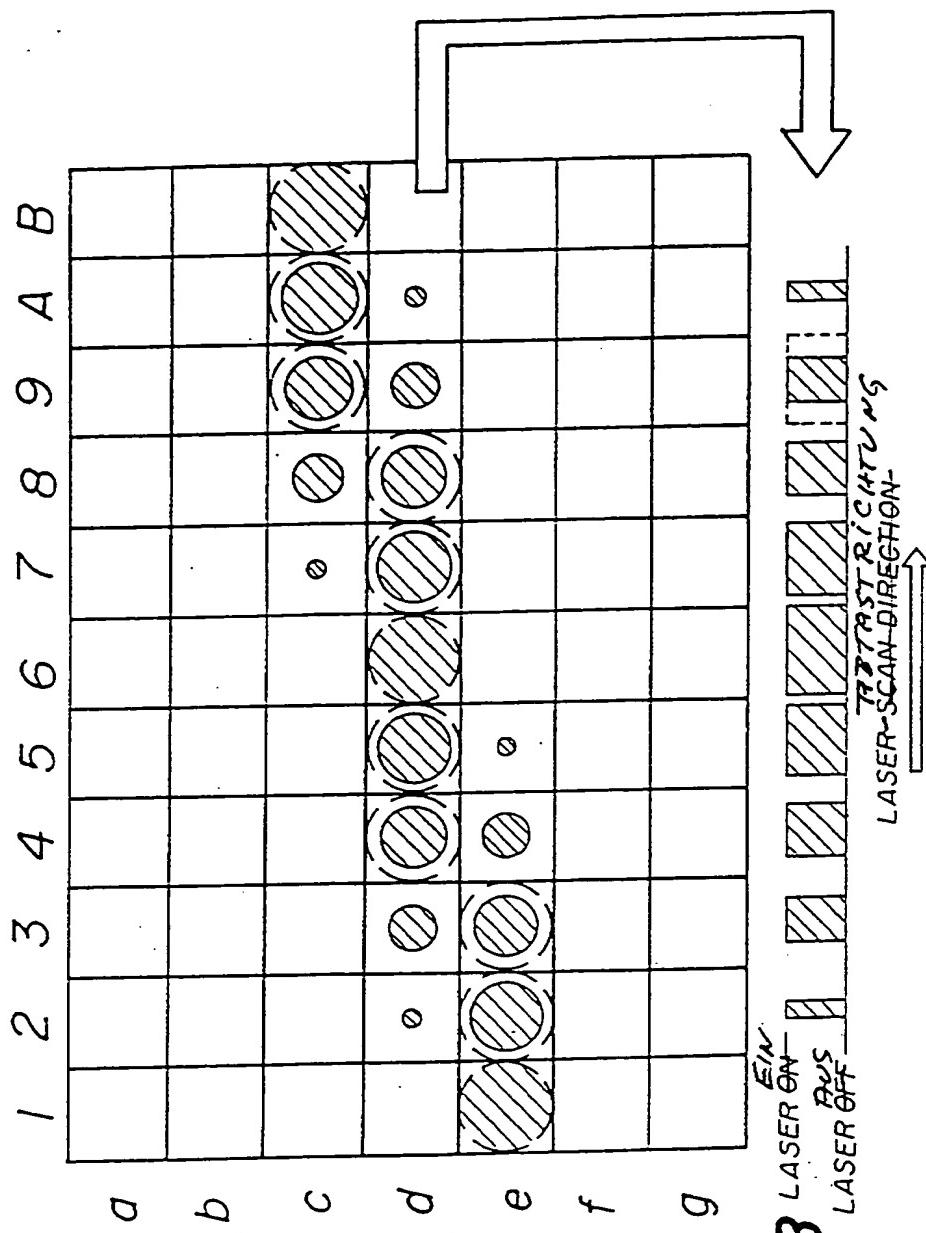


FIG. 42B

EIN LASER ON  
LASER BUS  
LASER OFF

STRIP RICHTUNG

FIG. 43A

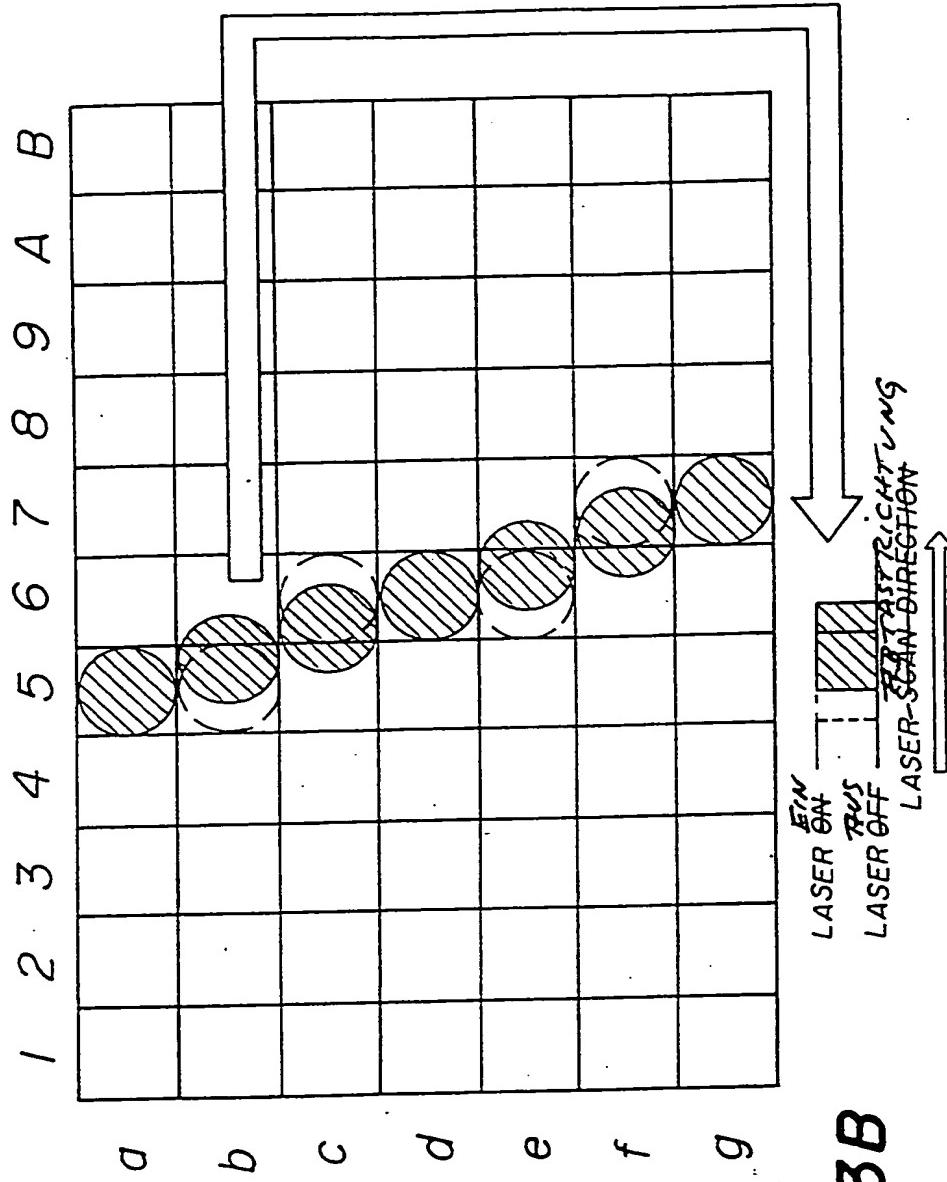


FIG. 43B

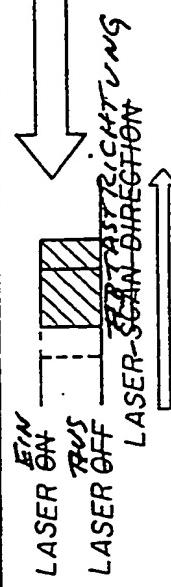


FIG.44A

SIGNAL	FIG.37	FIG.38
H/V	I	O
DIRI DIRO	O I	I O
B/W	O	I
U/L	O	-
GST	I	I
RUC	I	I
LLC	O	O
CCI CCO	I (2) O	I (2) O
RUASI RUASO	O O	I O
LLASI LLASO	- -	- -

FIG.44B

SIGNAL	FIG.37	FIG.38
RUCN2	O	O
RUCN1	I (3)	O (1)
RUCNO	I	I
RUDIRI RUDIRO	O I	I O
LLCN2	-	-
LLCN1	-	-
LLCNO	-	-
LLDIRI LLDIRO	-	-

FIG.44C

SIGNAL	FIG.37	FIG.38
G3	O	O
G3	I (5)	O (3)
G1	O	I
GO	I	I
NICHT-FRAN- AC-MATCH PASSEN	O	O

FIG.44D

SIGNAL	FIG.37	FIG.38
P3	O	O
P2	O (1)	O (1)
P1	O	O
P0	I	I